DYNAMIC FUNCTIONAL CONNECTIVITY IN FUNCTIONAL MAGNETIC RESONANCE IMAGING DATA

by

Yutting Xu

A dissertation submitted to The Johns Hopkins University in conformity with the requirements for the degree of Doctor of Philosophy.

Baltimore, Maryland

January, 2017

Abstract

Functional connectivity (FC) is defined as the undirected association between two or more functional magnetic resonance imaging (fMRI) time series. In the past decade, the assessment of FC using fMRI data has proven a powerful method for studying the functional organization of the brain. In recent years, a great deal of focus has shifted from the analysis of FC averaged over the duration of a fMRI scanning session, towards the analysis of dynamic (time-varying) changes in FC within a session. Here the goal is to study changes in FC on a time-scale ranging from seconds to minutes. It is now believed that studying dynamic FC can help provide insight into the fundamental mechanisms of brain function. Thus, the development of proper statistical methods for assessing these changes is of utmost importance. In this work, we present novel statistical methods for modeling dynamic FC in high-dimensional fMRI data.

In the first project, we develop the Dynamic Connectivity Detection (DCD) algorithm, which is a data-driven technique for detecting temporal change points in FC, and estimating the connectivity between a set of brain regions of interest (ROIs) for data within each segment defined by the change points. While the focus of this project is detecting state-related changes in FC, the second and third projects focus on modeling smoothly varying brain networks. In the second project, we introduce the multivariate volatility models developed in finance literature to the neuroimaging field. In particular, we focus on the Dynamic Conditional Correlation (DCC) model, and demonstrate its superior performance over other commonly used methods in estimating dynamic bivariate correlations in resting-state fMRI. However, when applying the DCC model in higher-dimensional settings, we encounter difficulties with biased estimation and large computational cost. Therefore, in the third project, we proposed a moment-based estimating method for the DCC model (MDCC), which achieves more accurate estimation and higher computational efficiency for high-dimensional fMRI time series. We compare the performance of the MDCC estimator with alternative estimators through a series of simulation studies and real data application, which successfully demonstrate the strength of proposed method.

Advisor:

Martin Lindquist, PhD

Committee:

James Pekar, PhD (chair, SOM radiology)

Martin Lindquist, PhD (advisor, SPH biostatistics)

Brian Caffo, PhD (SPH biostatistics)

Michelle Carlson, PhD (SPH mental health)

Alternates:

Vadim Zipunnikov, PhD (SPH biostatistics)

Alan Scott, PhD (SPH molecular microbiology and immunology)

Acknowledgments

First and foremost, I would like to thank my advisor, Dr. Martin Lindquist, for his continuous guidance and encouragement throughout graduate school. He was always showing a great interest in any of my progress, offering inspiring and thoughtful advice, expressing his confidence in me, and supporting me in all career pursuits. Whenever I focus too much on the deficiency in research, he always helped me to become optimistic about future with great patience. This dissertation would not have been possible without his support and help.

I would like to express my sincere gratitude to Dr. Vadim Zipunnikov, who has not only been a mentor for our collaborated project but also a great friend. I was deeply inspired by his dedication and passion in research, and truly appreciate his help and advice that helped me go through the difficult time during graduate study.

I would also like to give sincere thanks to all my dissertation committee and oral exam committee members, Dr. James Pekar, Dr. Brian Caffo, Dr. Michelle Carlson, Dr. Alan Scott, Dr. Elizabeth Ogburn and Dr. Joshua Vogelstein, for their valuable advice and help over the years.

ACKNOWLEDGMENTS

I am also grateful to all the HCP and SMART group members for their very helpful discussions and collaborations. It is so great to be part of the group which stimulates lots of interesting conversations about neuroimaging.

Special thanks to my fellow students and friends in the biostatistics department. It would be a lonely journey without their friendship.

Last but not least, I would like to thank my family, for their love and support.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Background

Functional Magnetic Resonance Imaging (fMRI) is a non-invasive technique for studying physiological changes associated with brain activation (Ogawa et al., 1992; Kwong et al., 1992). It measures changes in blood oxygenation and blood flow related to neuronal activity, and allows one to study how the brain functions, either in response to a certain task or when at rest. During the past 25 years, fMRI has provided researchers with an unprecedented access to the inner workings of the human brain, and this has led to countless new insights into how the brain processes information.

The data acquired in an fMRI experiment is both large and complex. During the course of an experiment, anywhere between 100-1000 brain volumes are sequentially acquired at fixed time points (typically 2 sec apart). Each volume consists of a number of uniformly spaced volume elements, or voxels, that partition the brain into a large number of equally sized boxes. A typical brain volume consists of roughly 100,000 voxels, and this constitutes the basic measurement in an fMRI experiment. In addition, the same experiment may be repeated several times for multiple subjects to facilitate population inference. Thus, each experiment produces a massive amount of high-dimensional data, creating substantial challenges for the statistical analysis of fMRI data.

There are a number of objectives in the analysis of fMRI data. These include: (i) localizing areas of the brain activated by a certain task; (ii) determining distributed networks that correspond to brain function; and (iii) making predictions about psychological or disease states. Dealing with these objectives is fundamentally a statistical problem, and the statistical analysis of fMRI data involves working with massive data sets that exhibit a complicated spatial and temporal noise structure. The size and complexity of the data make it difficult to create a full statistical model for describing its behavior, and a number of shortcuts are typically required to balance computational feasibility with model efficiency.

In this thesis we focus on the second objective, related to determining distributed networks that correspond to brain function. In particular, we are interested in methods for assessing functional connectivity (FC), or the undirected association between two or more fMRI time series (Friston, 2011). Here the time series can be obtained either from individual voxels, (Handwerker et al., 2012; Hutchison et al., 2013; Leonardi and Van De Ville, 2015), averaged over pre-specified regions of interest (Chang and Glover, 2010), or estimated using independent component analysis (Allen et al., 2012). The study of FC allows for the creation of functional networks, which are graphs of temporally correlated brain regions with distinct spatial distribution.

Previously, most fMRI studies have assumed a static network across the course of a scanning session, with constant FC between brain regions. However, recently evidence from a number of experimental studies (e.g., (Hutchison and Morton, 2015; Damaraju et al., 2014; Rashid et al., 2014)) have suggested that there exist dynamic changes in FC on time scales ranging from seconds to minutes, even if subjects are at rest. Thus, there has recently been increased interest in investigating possible dynamic changes in FC during the course of an fMRI experiment.

The statistical analysis of dynamic FC is a fundamentally high-dimensional problem with complicated spatial and temporal correlation structure. Detecting reliable, neuronally relevant changes in FC is difficult due to the low signal-to-noise ratio, physiological artifacts and variation in signal mean and variance over time. It is often unclear whether observed fluctuations in FC should be attributed to neuronal activity or noise (Lindquist et al., 2014). In addition, there remains uncertainty regarding the appropriate analysis strategy to use and how to interpret results (Hutchison et al., 2013; Calhoun et al., 2014). This thesis is focused on developing efficient and computationally scalable methods for assessing dynamic FC in high-dimensional settings.

1.2 Organizational Overview

In this thesis, we introduce several novel computational methods for analyzing the dynamic functional connectivity in fMRI data.

In Chapter 2, we introduce the Dynamic Connectivity Detection (DCD) algorithm, which is a data-driven technique for detecting temporal change points in functional connectivity, and estimating graphs representing connectivity within each partition between change points. DCD builds upon the framework of the recently developed Dynamic Connectivity Regression (DCR) algorithm (Cribben et al., 2012, 2013), which has proven to be an efficient technique for detecting changes in connectivity for problems consisting of a small to medium (< 50) number of regions. However, the technique runs into computational problems as the number of regions becomes large (> 100). To circumvent this problem we introduce DCD, which is faster, requires less user input, and is better able to handle high-dimensional data. It overcomes the shortcomings of DCR by adopting a simplified sparse matrix estimation approach and a different hypothesis testing procedure to determine change points. The application of DCD to simulated data, as well as fMRI data, illustrates the efficacy of the proposed method.

While DCD focuses on detecting abrupt "state-related" changes in FC, the methods developed in Chapters 3 and 4 focus on detecting more gradual changes in dynamic FC. In Chapter 3, we deal with the specific problem of estimating the dynamic behavior of pair-wise correlations between time courses extracted from two different regions of the brain. We critique the commonly used techniques for performing this type of analysis (such as sliding-windows correlation), and discuss some alternative methods used to model volatility in the finance literature that could also prove useful in the neuroimaging setting. In particular, we focus on the Dynamic Conditional Correlation (DCC) model (Engle, 2002a), which provides a model-based approach towards estimating dynamic correlations. We investigate the properties of several techniques in a series of simulation studies and find that DCC achieves the best overall balance between sensitivity and specificity in detecting dynamic changes in correlations. Finally, we illustrate its performance in an application to test-retest resting state fMRI data.

Finally, in Chapter 4 we propose a moment-based estimator for DCC (denoted MDCC), along with a fast estimation algorithm, which achieves both more accurate estimation and higher efficiency for high-dimensional time series. We investigate the properties of the newly proposed estimator in various simulation settings, and compare its performance with a recently developed composite likelihood method (Pakel et al., 2014). The application of MDCC to simulated and real resting state fMRI data, illustrates its efficacy.

1.3 Software

We provide the MATLAB toolbox along with the first and third projects, which are publicly available online (http://biostat.jhsph.edu/∼yuxu/), to facilitate future research in this field.

Chapter 2

Dynamic Connectivity Detection: an algorithm for determining functional connectivity change points in fMRI data

CHAPTER 2. DYNAMIC CONNECTIVITY DETECTION

Abstract

Recently there has been an increased interest in using fMRI data to study the dynamic nature of brain connectivity. In this setting, the activity in a set of regions of interest (ROIs) is often modeled using a multivariate Gaussian distribution, with a mean vector and covariance matrix that are allowed to vary as the experiment progresses, representing changing brain states. In this work, we introduce the Dynamic Connectivity Detection (DCD) algorithm, which is a data-driven technique to detect temporal change points in functional connectivity, and estimate a graph between ROIs for data within each segment defined by the change points. DCD builds upon the framework of the recently developed Dynamic Connectivity Regression (DCR) algorithm, which has proven efficient at detecting changes in connectivity for problems consisting of a small to medium (< 50) number of regions, but which runs into computational problems as the number of regions becomes large (> 100). The newly proposed DCD method is faster, requires less user input, and is better able to handle high-dimensional data. It overcomes the shortcomings of DCR by adopting a simplified sparse matrix estimation approach and a different hypothesis testing procedure to determine change points. The application of DCD to simulated data, as well as fMRI data, illustrates the efficacy of the proposed method.

2.1 Introduction

Functional connectivity (FC) is the study of the temporal dependencies between distinct, possibly spatially remote, brain regions (Friston, 1994). Assessing FC using functional Magnetic Resonance Imaging (fMRI) data, has proven particularly useful for discovering patterns indicting how brain regions are related, and comparing these patterns across groups of subjects (Friston, 2011; Lindquist, 2008). In recent years, it has become one of the most active research areas in the neuroimaging community, and it is a central concept in the long term goal of understanding the human connectome (Sporns et al., 2005). The hope is that increased knowledge of networks and connections will help facilitate research into a number of common brain disorders.

FC is fundamentally a statistical concept, and is typically assessed using statistical measures such as correlation (Biswal et al., 1995), cross-coherence (Sun et al., 2004), and mutual information (Jeong et al., 2001). In the past few years it has become increasingly common to assume that the fMRI time series data follows a multivariate Gaussian distribution, and quantify FC using the estimated covariance, correlation or precision (inverse covariance) matrix (Varoquaux et al., 2010; Cribben et al., 2012, 2013). In this setting there is a well-known relationship between the estimated precision matrix and the underlying network graph of interest, and the use of algorithms for estimating sparse precision matrices (and thus graphs) have become critical (Friedman et al., 2008).

Most functional connectivity analyses performed to date have generally assumed that the relationship within functional networks is stationary across time. However, in recent years there has been an increased interest in studying dynamic changes in FC over time. These analyses have shown that rather than being static, functional networks appear to fluctuate on a time scale ranging from seconds to minutes (Chang and Glover, 2010). Here changes in both the strength and directionality of functional connections have been observed to vary across experimental runs (Hutchison et al., 2013), and it is believed that these changes may provide insight into the fundamental properties of brain networks.

When the precise timing and duration of state-related changes in FC are known before hand, it is possible to apply methods such as the psychophysiological interactions (PPI) technique (Friston et al., 1997) or statistical parametric networks analysis (Ginestet and Simmons, 2011). However, in many research settings the nature of the psychological processes being studied is unknown, particularly in resting-state fMRI (rfMRI), and it is therefore important to develop methods that can describe the dynamic behavior in connectivity without requiring prior knowledge of the experimental design. In the past couple of years, a number of such approaches have been suggested in the neuroimaging literature, including the use of sliding window correlations (Chang and Glover, 2010; Allen et al., 2012; Hutchison et al., 2013; Handwerker et al., 2012), change point models (Cribben et al., 2012, 2013), and volatility models (Lindquist et al., 2014).

One example is dynamic connectivity regression (DCR), which is a data-driven technique for partitioning a time course into segments and estimating the different connectivity networks within each segment (Cribben et al., 2012). It applies a greedy search strategy to identify possible changes in FC using the Bayesian Information Criteria (BIC). While optimizing the BIC value within each subsequence, DCR utilizes the GLASSO algorithm to estimate a sparse inverse covariance matrix. This is followed by a secondary analysis of the candidate split points, where a permutation test is performed to decide whether or not the reduction in BIC at that time point is significant enough to be deemed a true change point. The structure of the DCR algorithm is briefly demonstrated in Fig. 2.1.

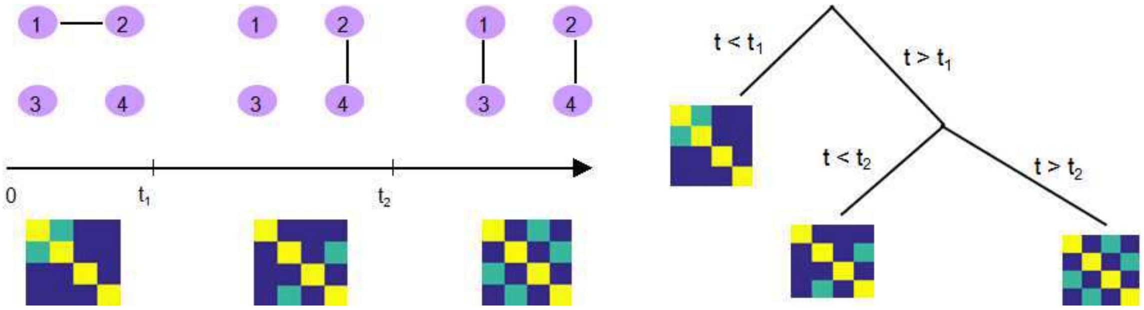

Figure 2.1: An illustration of DCR. (Left) There exist two change points t_1 and t_2 where the connectivity between 4 ROIs changes as shown in the corresponding precision matrix. (Right) DCR discovers the change points, recursively, using a binary search tree.

While the DCR algorithm has proven useful for detecting changes in FC, it has two major drawbacks. First, the computational cost of the algorithm increases rapidly with the number of ROIs. As the number of ROIs surpasses 50, the computation time can become prohibitive. Second, DCR requires a number of user-specified input parameters, some of which may be difficult to optimize without in-depth knowledge of the experiment and familiarity with the algorithm.

In this work, we introduce the Dynamic Connectivity Detection (DCD) algorithm for change point detection in fMRI time series data, as well as the estimation of a graph representing connectivity within each partition. It builds upon the basic DCR framework, using the same binary search tree structure to recursively identify potential change points. However, it replaces a number of critical components of DCR, including the manner in which the sparse matrix estimation is performed and significant change points determined. An adaptive thresholding approach is used to estimate a sparse covariance matrix, which provides a significant speed up in computation time compared to the GLASSO algorithm, and improves scalability. In addition, the permutation test used to detect significant change points is replaced by an alternative hypothesis test. Because of these changes, all the input parameters in the DCD algorithm have a clear interpretation in the context of hypothesis testing, allowing users to specify the desired control of Type I and Type II errors.

This paper is organized as follows. In Section 2 we begin by briefly reviewing the basic steps of DCR, followed by a discussion of sparse parameter estimation, and a description of the new DCD algorithm for single-subject change point detection and graph estimation. Thereafter we demonstrate the performance of DCD in Sections 3 and 4 by applying the method to a series of simulation studies and experimental data. The obtained results are contrasted with similar results obtained using DCR.

The paper concludes with a discussion.

2.2 Methods

Consider fMRI data from a single subject consisting of multivariate time series, where each dimension corresponds to activity from a single region of interest (ROI). Assume that the measurement vector at each time point follows a multivariate Gaussian distribution, whose parameters may vary across time. Throughout, we denote the measurement at time t as $\mathbf{y}(t)$ $(1 \leq t \leq T)$, which represents a J-dimensional Gaussian random vector whose distribution is $\mathcal{N}(\boldsymbol{\mu}(t), \Sigma(t))$.

The goal of Dynamic Connectivity Detection (DCD) is to detect temporal change points in functional connectivity and estimate a sparse connectivity graph for each segment, where the vertices are ROIs and the edges represent the relationship between ROIs. More specificity, we seek to partition the time series into several distinct segments, within which the data follows a multivariate Gaussian distribution with a different mean vector or covariance matrix from its neighboring segments. Further, for each segment we seek to estimate a graph representing connectivity between ROIs in the segment.

The DCR algorithm (Cribben et al., 2012, 2013) was previously developed to deal with the same problem. While, DCR has proven efficient at detecting changes in connectivity for problems consisting of a small to medium (less than 50) number of regions, it runs into computational problems as the number of regions becomes large (more than 100). The proposed DCD algorithm seeks to circumvent these issues by updating how (i) the underlying mechanisms by which change points are determined, and (ii) network structures are identified. Before discussing DCD in detail, we begin by giving a brief overview of DCR and sparse parameter estimation.

2.2.1 Dynamic Connectivity Regression (DCR)

The original DCR algorithm (Cribben et al., 2012), dealt with detecting change points in a group of subjects, but here we concentrate on the single subject case (Cribben et al., 2013). DCR aims at detecting temporal change points in functional connectivity and estimating a graph of the conditional dependencies between ROIs, for data that falls between each pair of change points. The measured signal is modelled as a Gaussian random vector where each element represents the activity of one region. The partitions in DCR are found using a regression tree approach. It attempts to first identify a candidate change point using the Bayesian Information Criterion (BIC), and then perform a permutation test to decide whether it is significant. If a significant change points is found, the same procedure is recursively applied to search for more changes points by further splitting the subset; see Fig. 1 for an illustration.

The required user specified input parameters for the algorithm are:

1. Δ: the minimum possible distance between adjacent changes points, chosen based on prior knowledge about the fMRI experiment.

2. $\lambda-$*list*: the full regularization path of tuning parameters λ required by GLASSO.
3. ξ: the mean block size of the stationary bootstrap.
4. α: the significance level for the permutation test.
5. N_b : the number of bootstrap samples.

Suppose we have a J-dimensional time series $\mathbf{Y} := \{\mathbf{y}(t)\}_{1\leq t\leq T}$, where the $\mathbf{y}(t)'s$ are assumed to be independent identically distributed random variables which follow a multivariate Gaussian distribution. Here the mean vector can be estimated using the sample mean, and a sparse precision matrix can be estimated using the GLASSO technique (see next section for more detail). In order to choose the appropriate tuning parameter λ needed for GLASSO, the full regularization path $\lambda - list$ is run, and the optimal value is selected based on the value that minimizes the BIC. Finally, the model is refit without regularization, but keeping the zero elements fixed, and the optimized baseline BIC for the original time series, b_0, is recorded.

For all possible split points t ($\Delta \leq t \leq T-\Delta$), the same procedure is repeated, and the BIC score for the two subsequences $\mathbf{Y}_1 := \{\mathbf{y}(t')\}_{1\leq t'\leq t}$ and $\mathbf{Y}_2 := \{\mathbf{y}(t')\}_{t+1\leq t'\leq T}$, denoted $b_1(t)$ and $b_2(t)$, respectively, are computed. A time point t_0 is chosen as a candidate change point, if it (i) produces the smallest combined BIC score $b_1(t_0) + b_2(t_0)$ for all possible split points t, and (ii) the combined BIC score is smaller than the baseline b_0. In the continuation we let $\delta_b = b_0 - (b_1(t_0) + b_2(t_0))$ represent the decrease in BIC at t_0.

Because change points are defined by a decrease in BIC, a random permutation procedure is used to create a $100(1-\alpha)\%$ confidence interval for BIC reduction at the candidate change point t_0, to determine whether it should be deemed a significant change point. Using a stationary bootstrap procedure with mean block size ξ, permuted time series are repeatedly created. Each time course is partitioned at time t_0 and the BIC reduction is computed as described above. The procedure is performed N_b times, thus allowing for the creation of a permutation distribution for the BIC reduction. If δ_b is more extreme than the $(1-\alpha)$ quantile of the permutation distribution, we conclude t_0 is a significant change point. This procedure is recursively applied to each individual partition until no further split reduces the BIC score.

2.2.2 Sparse Parameter Estimation

The estimation of the covariance and precision matrix is a critical step in identifying candidate change points in the DCR algorithm. While the number of ROIs J is moderate, and the length of time series T is large, the sample covariance matrix S is a consistent estimator of the covariance matrix Σ. However, in high dimensional settings, when J is large compared to the sample size T, S has an infinite determinant, leading to divergence in the numerical algorithm. Thus sparsity constraints are required to estimate the covariance, or precision matrix, consistently.

In this section we discuss two methods for performing sparse matrix estimation. While the original DCR method imposes sparsity on the precision matrix, the proposed DCD algorithm instead seeks to estimate a sparse covariance matrix. By making this shift, we can use a newly developed adaptive thresholding approach that provides a faster, more scalable solution to the change point problem described above. Statistically this changes the interpretation of the problem, as zeros in the precision matrix correspond to conditional independence between variables, while zeros in a covariance matrix correspond to marginal independence between variables. In a series of simulations and an application to real data we examine the implications of this choice.

2.2.2.1 Graphical LASSO (GLASSO)

The Least Absolute Shrinkage and Selection Operator (LASSO) technique (Tibshirani, 1996), is often used for shrinkage and feature selection in regression problems. It adds an L_1 penalty term to the objective function, thus producing more interpretable models with some coefficients forced to be exactly zero. The Graphical LASSO (GLASSO) (Friedman et al., 2008) is an extension of this idea to graphical models, aimed at estimating sparse precision matrices. Based on the assumption that the observed data vectors $\{\mathbf{y}(t)\}_{1\leq t\leq T}$ follow a multivariate Gaussian distribution with covariance matrix Σ, it adds an L_1 norm penalty to the elements of the precision matrix $\Omega = \Sigma^{-1}$, and estimates the mean vector $\boldsymbol{\mu}$ and precision matrix Ω by maximizing the penalized log-likelihood. After substituting the sample mean (the

MLE of $\boldsymbol{\mu}$) into the objective function, this reduces to:

$$\log det(\Omega) - tr(S\Omega) - \lambda \|\Omega\|_1$$

where S is the empirical covariance matrix, and the parameter λ controls the amount of regularization. Maximizing the penalized profile log-likelihood gives a sparse estimate of Ω.

If the ij^{th} element of matrix Ω is zero, the variables $y_i(t)$ and $y_j(t)$ are conditionally independent, given the other variables. We can therefore define a connectivity graph $G = (V, E)$ with the ROIs the vertices in V, and prune the edge between vertices i and j if the variables are conditionally independent. Thus increasing the sparsity of Ω provides a sparser graphical representation of the relationship between the variables.

2.2.2.2 Adaptive Thresholding Approach

Here we introduce an adaptive thresholding approach that allows one to estimate a sparse covariance matrix. Again, assume the data $\{\mathbf{y}(t)\}_{1 \leq t \leq T}$ follows an i.i.d. multivariate Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}, \Sigma)$. In this setting, the sample mean

$$\hat{\boldsymbol{\mu}} = \frac{1}{T} \sum_{1 \leq t \leq T} \mathbf{y}(t)$$

is a consistent estimator of $\hat{\boldsymbol{\mu}}$.

To estimate the covariance matrix, we begin by using the empirical covariance matrix

$$\hat{\Sigma} = \frac{1}{T} \sum_{1 \leq t \leq T} (\mathbf{y}(t) - \hat{\boldsymbol{\mu}})^T (\mathbf{y}(t) - \hat{\boldsymbol{\mu}})$$

as a candidate estimator of Σ. To achieve sparsity we investigate whether individual elements should be set equal to zero following an idea of Cai et al. (Cai and Liu, 2011), where a method to model the distribution of $\hat{\Sigma}_{ij}$ is proposed.

Let $X_t^{ij} := (y_i(t) - \mu_i)(y_j(t) - \mu_j)$, where a subscript represents a single dimension of a vector, then the ij^{th} element of $\hat{\Sigma}$ is:

$$\hat{\Sigma}_{ij} = \frac{1}{T} \sum_{1 \leq t \leq T} X_t^{ij} = \bar{X}^{ij} \tag{2.1}$$

Now $X_1^{ij}, X_2^{ij}, ...X_T^{ij}$ is a sequence of i.i.d. random variables with $E[X_t^{ij}] = E[(y_i(t) - \mu_i)(y_j(t) - \mu_j)] = \Sigma_{ij}$ by definition, and further assume $Var[X_t^{ij}] = \delta_{ij}^2 < \infty$. Then by the Central Limit Theorem,

$$\sqrt{T}\big(\hat{\Sigma}_{ij} - \Sigma_{ij}\big) \to \mathcal{N}(0, \delta_{ij}^2)$$

A natural estimate of δ_{ij}^2 is given by:

$$\hat{\delta_{ij}^2} = \frac{1}{T} \sum_{1 \leq t \leq T} (X_t^{ij} - \bar{X}^{ij})^2 \tag{2.2}$$

Alternatively, one can use the Jackknife technique to estimate the variance of estimator $\hat{\Sigma}_{ij}$ directly (see Appendix A1.2).

Using this result, we can test $H_0 : \Sigma_{ij} = 0$ $vs.$ $H_1 : \Sigma_{ij} \neq 0$ at significance level η as follows:

$$|\frac{\sqrt{T}\hat{\Sigma}_{ij}}{\hat{\delta_{ij}}}| = \frac{T|\hat{\Sigma}_{ij}|}{\sqrt{\sum_{t=1}^{T}(X_t^{ij} - \bar{X}^{ij})^2}} > z_{1-\eta/2}$$

If we successfully reject the null hypothesis, we can conclude that $\Sigma_{ij} \neq 0$ and keep $\hat{\Sigma}_{ij}$ as the estimator for Σ_{ij}. Otherwise we modify the candidate estimator and set $\hat{\Sigma}_{ij} = 0$. Similarly, using the diagonal elements of $\hat{\Sigma}$ as estimates of the variance of $\hat{\boldsymbol{\mu}}$, we can perform a hypothesis testing for each element of $\boldsymbol{\mu}$ and obtain a sparse estimate of $\hat{\boldsymbol{\mu}}$. Since the testing procedure is performed for a potentially large number of parameters, we need to correct for multiple comparisons (Lindquist and Mejia, 2015).

2.2.3 Dynamic Connectivity Detection (DCD)

The Dynamic Connectivity Detection (DCD) algorithm seeks to speed up the DCR algorithm, while achieving equivalent, or improved, results. The general procedure of DCD is similar to DCR, where a candidate split point is identified based on whether it further maximizes a likelihood-based function, and a hypothesis test is performed to decide whether this candidate split point is statistically significant. If a significant change point is found, the procedure is applied recursively to each of the two subsequences in order to find further split points.

The major improvement from DCR to DCD is that we incorporate the adaptive thresholding approach as our sparse matrix estimation method, which successfully improves upon the computational efficiency. In addition, during each step, a binary 'mask' representing the non-zero parameter elements (in the mean vector and covariance matrix) is saved for each partition. If an additional change point is found for this partition, the 'mask' is imposed on the parameters of both 'child' partitions (the two subsets of time series created by splitting the data at the change point). This implies that if the estimate of one element of the covariance matrix for some partition is zero, then the estimate of corresponding element in any sub-partition will also be zero. The recursive sparsity feature is illustrated in Fig 2.2.

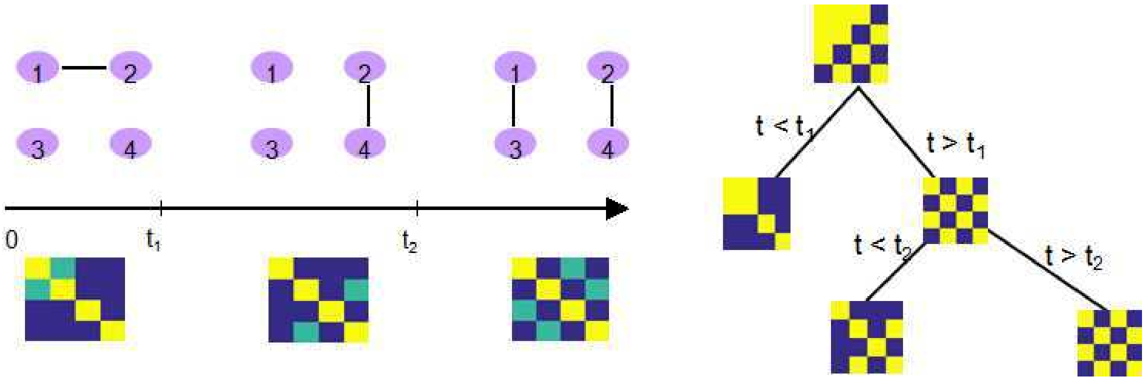

Figure 2.2: An illustration of how the sparsity structure is recorded in DCD. (Left) The split points t_1 and t_2, and the corresponding covariance matrix within each partition. Yellow elements in the covariance matrix plot represents 1, green elements 0.5, and blue elements 0. (Right) DCD uses a binary mask to record the sparsity structure at each node of the binary search tree.

All input parameters in DCD have a clear statistical interpretation, enhancing its user-friendliness. The required user specified input parameters for the algorithm are:

1. α: the type I error bounds for the hypothesis tests used to determine significant

splits.

2. β: the type II error bounds for the hypothesis test used to determine significant splits.
3. η: the type I error bound for the hypothesis test used to determine the sparsity of the covariance matrix.

Since the length of the time series partition affects statistical inference, we need to calculate the minimum partition length Δ needed to achieve the desired error bounds. We apply a power analysis based on a two sample t-test to calculate Δ from the inputs α and β; for details please refer to Appendix A1.1.

Given a J-dimensional time series $\mathbf{Y} := \{\mathbf{y}(t)\}_{1\leq t\leq T}$, we begin by calculating the maximized baseline log-likelihood L_0 under the assumption that

$$\mathbf{y}(t) \quad \overset{i.i.d}{\sim} \quad \mathcal{N}(\boldsymbol{\mu}_0, \Sigma_0), \qquad 1 \leq t \leq T.$$

Hence, the log-likelihood function is given by

$$L(\boldsymbol{\mu}_0, \Sigma_0|\mathbf{Y}) \propto -\sum_{t=1}^{T}(\mathbf{y}(t) - \boldsymbol{\mu}_0)^T\Sigma_0^{-1}(\mathbf{y}(t) - \boldsymbol{\mu}_0) - T\log(\det\Sigma_0). \tag{2.3}$$

We first calculate the sample mean and sample covariance matrix as the maximum likelihood estimator of $\boldsymbol{\mu}_0$ and Σ_0, and then further improve the estimator by performing the adaptive thresholding method described in Section 2.2.2.2, in order to obtain a sparse mean vector $\hat{\boldsymbol{\mu}}_0$ and sparse covariance matrix $\hat{\Sigma}_0$.

The maximized log-likelihood function can now be expressed as:

$$L_0 = -T\Big(tr(\hat{\Sigma}_0^{-1}S) + \log(\det \hat{\Sigma}_0)\Big)$$

where S is the normalized scatter matrix:

$$S = \frac{1}{T}\sum_{1\leq t\leq T}(\mathbf{y}(t) - \hat{\boldsymbol{\mu}}_0)^T(\mathbf{y}(t) - \hat{\boldsymbol{\mu}}_0)$$

While calculating the sparse structure of parameter $\boldsymbol{\theta}_0 = (\boldsymbol{\mu}_0, vec\{\Sigma_0\})$, a binary array *mask* is saved, indicating the non-zero elements of θ_0. It is assumed that any subsequence of the time series will satisfy the parent sparsity property.

For any possible candidate split point t ($\Delta \leq t \leq T - \Delta$), assume the two subsequences $\mathbf{Y}_1 := \{\mathbf{y}(t')\}_{1\leq t'\leq t}$ and $\mathbf{Y}_2 := \{\mathbf{y}(t')\}_{t+1\leq t'\leq T}$ follow multivariate Gaussian distribution with parameters $\boldsymbol{\theta}_{1t} = (\boldsymbol{\mu}_{1t}, vec\{\Sigma_{1t}\})$ and $\boldsymbol{\theta}_{2t} = (\boldsymbol{\mu}_{2t}, vec\{\Sigma_{2t}\})$, respectively. Here only the upper triangular elements are used when vectorizing the covariance matrix. The dimension of the parameter vector is therefore $J + J * (J+1)/2 = (J+1)(J+2)/2$. Next, the maximum likelihood estimators $\hat{\boldsymbol{\theta}}_{it}^{ML}$ (i=1,2) are computed, imposing the parent sparsity structure by taking the Hadamard product with the *mask* vector:

$$\hat{\boldsymbol{\theta}}_{it} = \hat{\boldsymbol{\theta}}_{it}^{ML} \otimes mask, \qquad i = 1, 2$$

Now the maximized log-likelihood under current split point t can be obtained as follows:

$$L_t = L(\hat{\boldsymbol{\theta}}_{1t}|\mathbf{Y}_1) + L(\hat{\boldsymbol{\theta}}_{2t}|\mathbf{Y}_2).$$

Similar to DCR we can now step through all possible candidate split points and find the one, denoted t_0, which shows the maximum improvement in log-likelihood L_t compared to L_0:

$$t_0 = \underset{t}{argmax}(L_t - L_0)_+$$

If the maximum L_{t_0} is less than the baseline L_0, the DCD procedure returns no detected split points; otherwise a set of hypothesis tests are performed to determine whether t_0 is a significant change point.

For the sake of clarity, denote the Gaussian distribution parameters of the two subsequences as $\boldsymbol{\theta}_i = (\boldsymbol{\mu}_i, vec\{\Sigma_i\}) := \boldsymbol{\theta}_{it}$, (i = 1,2). We now seek to test:

$$H_{j0} : \boldsymbol{\theta}_1(j) = \boldsymbol{\theta}_2(j) \ vs. \ H_{j1} : \boldsymbol{\theta}_1(j) \neq \boldsymbol{\theta}_2(j), \qquad j \in \{j' : mask(j') = 1\} \tag{2.4}$$

If any of the non-zero parameters are significantly different for the two subsequences, i.e. if we reject any of the null hypotheses, then we conclude that t_0 is a significant change point for partitioning the time series $\mathbf{Y}$. We use Bonferroni correction to control the family-wise error rate (FWER), and reject H_{j0} if the p-value is less than $\frac{\alpha}{\sum_{j'} mask(j')}$.

To perform each test we use Welch's t test (two-sample t-test for unequal variance). For $j \leq J$, use the diagonal element of $\hat{\Sigma}$ as an estimate of the variance of $\hat{\boldsymbol{\mu}}$; and for $j > J$, use the estimator described in Eq. 2.2 to estimate the variance of each element of $\hat{\Sigma}$. If t_0 is identified as a significant change point, continue searching for more change points by recursively repeating the above procedure on the two 'child' subsequences until no further change points are returned; otherwise finish the DCD procedure by returning a null value.

The complete procedure for performing the DCD algorithm is summarized below:

1. Take the input parameters α, β, η, and calculate the minimum partition length Δ as described in Appendix A1.1.

2. Consider the full multivariate time series with length T, calculate the sparsity structure of its multivariate normal distribution parameters as described in Section 2.2.2.2, estimate the mean vector and a sparse covariance matrix accordingly, and calculate the baseline likelihood function L_0.

3. For each value of t ranging from Δ to $T - \Delta$, partition the time series into two subsequences $\{1 : t\}$ and $\{t + 1 : T\}$, calculate the sparsity structure of parameters based on the parent sparsity structure from Step (2), then calculate the combined likelihood function using the estimated sparse parameters.

4. Find the time point which produces the largest increase in combined likelihood function, perform the hypothesis test described in Eq. 2.4 to determine whether

it is a significant change point. If yes, split the time series into two partitions accordingly.

5. Apply Steps (2) - (4) recursively to each partition until no further change points are found.
6. After detecting all change points, estimate a connectivity graph for each partition using a sparse matrix estimation technique, such as Adaptive Thresholding Approach to obtain a covariance graph or GLASSO to obtain a connectivity graph.

2.3 Simulations

A series of simulations were performed to test the efficacy of the new DCD algorithm, and compare its performance to the DCR method. For this reason, we adopt simulation settings inspired by those found in the original DCR work (Cribben et al., 2012). However, in contrast to that work, for each simulation the connectivity pattern and strength between nodes remains the same across different subjects, since our focus is on the single subject case instead of on group-level inference. In addition, the object of each simulation in this paper is focused on identifying the timing of the connectivity change points, rather than explicitly assessing the quality of the estimation of the underlying graphs.

The descriptions and parameter settings for each simulation are listed below. Here

N, T and p represent the number of subjects, the length of the time series, and the number of regions, respectively. The true dependency between ROIs (i.e. the precision matrices) are shown as heat maps in Figs. 2.3 - 2.7. More details regarding the exact strength of these connections can be found in Appendix A1.3. Here the notation $(i, j) = k$ indicates that the (i, j) element of the precision matrix takes the value k. All unspecified diagonal elements are one and non-diagonal elements are zero. In the latter case, the ROIs were made up of i.i.d. Gaussian noise indicating a lack of functional connectivity. Hence, each simulation is created assuming sparsity in the precision matrix, which should theoretically benefit DCR over DCD, which imposes sparsity in the covariance matrix.

For each simulation, both the DCD and DCR approaches were applied to the N subjects individually. Since the DCR algorithm has many parameters, and according to previous work several are insensitive to change, we fix several of them as follows:

$$\Delta = 50,\ \lambda - list = (2^0, 2^{-1}, ..., 2^{-9}),\ N_b = 50,\ \xi = \Delta/2.$$

For DCD, we fix $\eta = 0.05$. All remaining parameters are altered depending on the simulation setting.

Below we list a brief description of each simulation study.

- **Simulation 1**

  Description: The data is white noise with no connectivity change points.

Size: $N = 20$, $T = 1000$, $p = 20$

DCD parameters: $(\alpha, \beta) = (0.05, 0.1)$;

DCR parameters: $\alpha = 0.05$.

- **Simulation 2**

  Description: There are two change points at times 200 and 400. Spikes are imposed onto the time series, imitating a common artifact found in fMRI data. For each subject there are 5 randomly placed spikes, each with magnitude 15.

  Size: $N = 20$, $T = 1000$, $p = 20$

  DCD parameters: $(\alpha, \beta) = (0.05, 0.1)$;

  DCR parameters: $\alpha = 0.05$.

- **Simulation 3**

  Description: There are three change points at times 125, 500 and 750.

  Size: $N = 15$, $T = 1000$, $p = 20$

  DCD parameters: $(\alpha, \beta) = (0.05, 0.05)$;

  DCR parameters: $\alpha = 0.05$.

- **Simulation 4**

  Description: There is a single change point at time 100.

  Size: $N = 25$, $T = 200$, $p = 5$

  DCD parameters: $(\alpha, \beta) = (0.05, 0.1)$;

DCR parameters: $\alpha = 0.05$.

- **Simulation 5**

  Description: There are five change points at times 200, 300, 500, 600, and 800.

  Size: $N = 20$, $T = 1000$, $p = 20$

  DCD parameters: $(\alpha, \beta) = (0.05, 0.05)$;

  DCR parameters: $\alpha = 0.05$.

- **Simulation 6**

  Description: There are four change points at times 200, 400, 600, and 800.

  Size: $N = 20$, $T = 1000$, $p = 20$

  DCD parameters: $(\alpha, \beta) = (0.05, 0.05)$;

  DCR parameters: $\alpha = 0.05$.

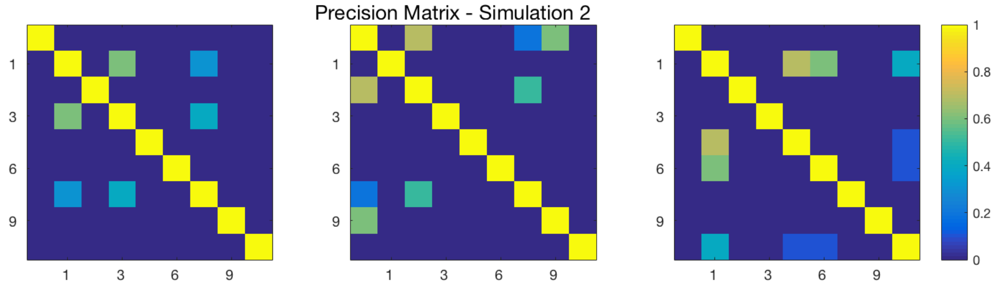

Figure 2.3: The dependency structure used in each of the three partitions of Simulation 2.

The results of the simulations are shown in Figs. 2.8 - 2.13. In each figure, the y-axis represents the subject number, while the x-axis represents time points. All red

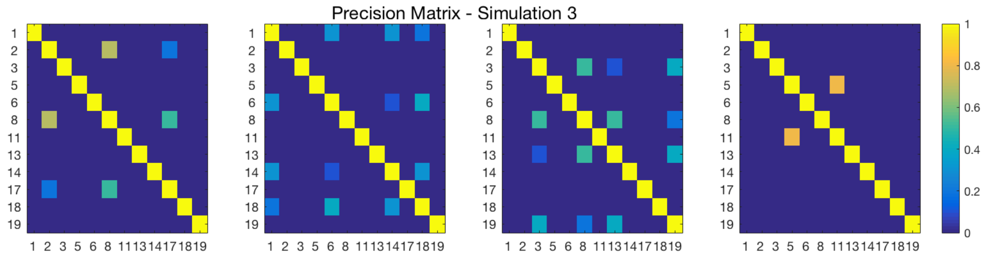

Figure 2.4: The dependency structure used in each of the four partitions of Simulation 3.

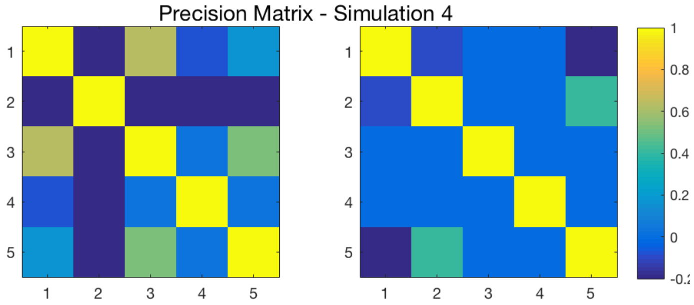

Figure 2.5: The dependency structure between regions $1 - 5$ (all other regions are conditionally independent) used in each of the two partitions of Simulation 4.

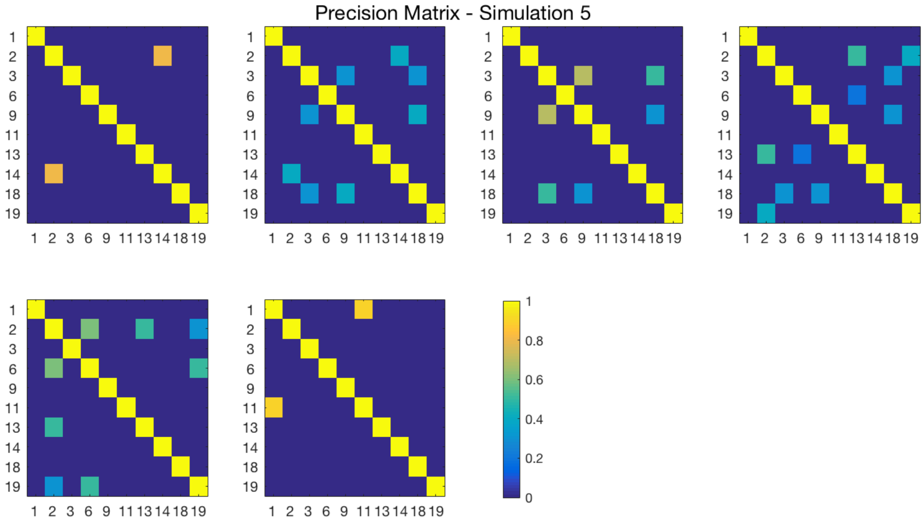

Figure 2.6: The dependency structure used in each of the six partitions of Simulation 5.

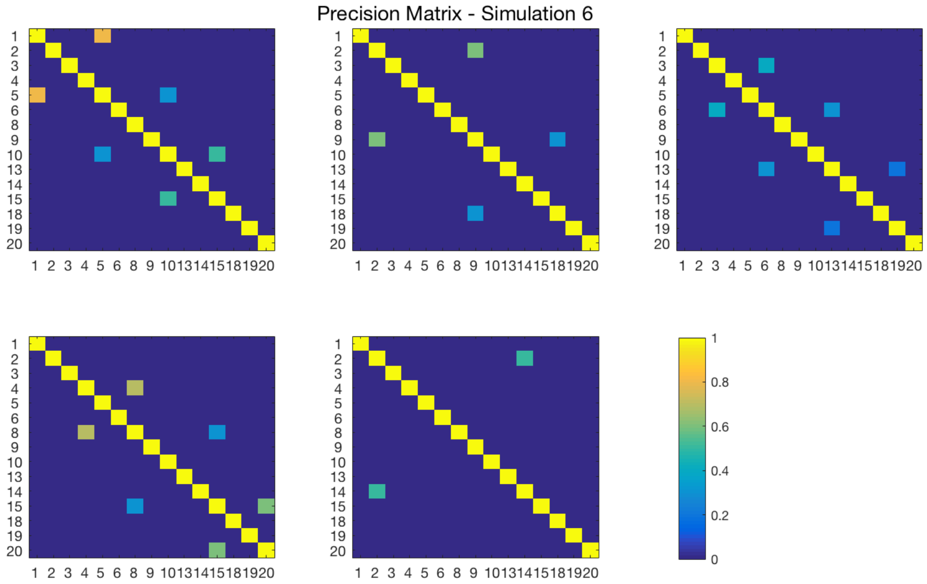

Figure 2.7: The dependency structure used in each of the five partitions of Simulation 6.

crosses in the left sub figures represent change points detected for each subject by DCD, and the blue circles are those detected by DCR. The blue vertical line indicates the true change points for each simulation setting. In Table 2.1, we list the respective runtimes of DCD and DCR for each simulation. The computing platform used was an Intel Core i5-3210M CPU 2.5 GHz with 16.0 GB RAM.

	Simulation 1	Simulation 2	Simulation 3	Simulation 4	Simulation 5	Simulation 6
DCR run time	249.2	351.5	408.5	13.3	585.5	581.1
DCD run time	8.0	5.8	12.4	0.7	16.9	18.1
ratio $= \frac{DCRtime}{DCDtime}$	31.1	61.1	33.0	18.6	34.5	32.1

Table 2.1: Runtime comparison between the DCD and DCR algorithms for each simulation. Runtime is measured in units of seconds.

The results of Simulation 1, where there are no true change points, are shown

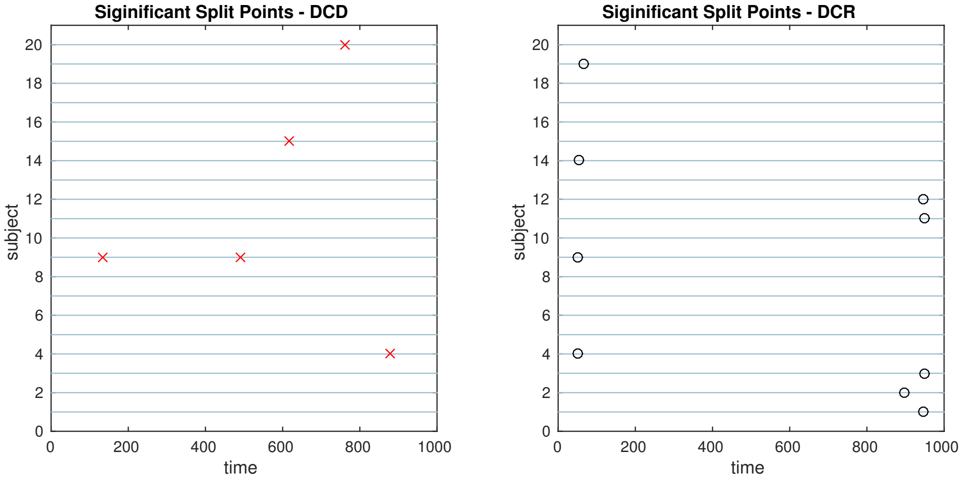

Figure 2.8: The results of Simulation 1. (Left) The red crosses show significant split points found by DCD. (Right) The blue circles show significant split points found by DCR. Here there should ideally be no change points for any of the subjects.

in Fig. 2.8. The DCD algorithm finds 5 false positive change points, whereas the DCR algorithm finds 9. Interestingly, the DCR false positives are primarily grouped at the time points Δ and $T-\Delta$. The reason for this is that when adding the BIC score from two sub-series of lengths n_1 and n_2, where $n_1+n_2=n$, and assuming the number of parameters $k_1 \approx k_2 \approx k$, the total penalty term is $klog(n_1)+klog(n_2) \propto log\big(n_1(n-n_1)\big)$, which favors small or large values of n_1 when minimizing the BIC. In addition, the runtime of DCD is approximately 30 times faster than DCR, providing a significant decrease in computation time.

The results of Simulation 2 are shown in Fig. 2.9. Here there exist two true change points, the first at time 200, and the second at time 400. In addition, there are 5 spikes placed at random time points for each subject. Both algorithms do a

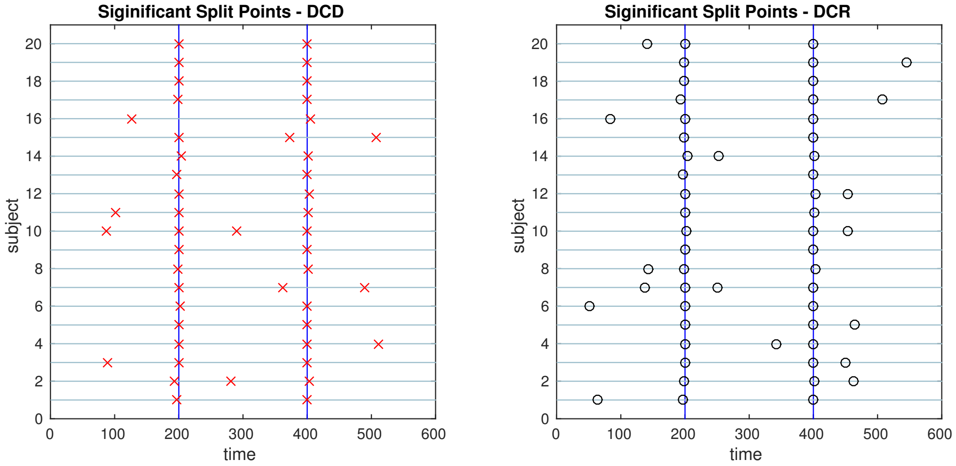

Figure 2.9: The results of Simulation 2. (Left) The red crosses show significant split points found by DCD. (Right) The blue circles show significant split points found by DCR. The blue vertical lines indicate the timing of the true change points.

good job of detecting the true change points in most cases, with a few instances of false positives for each. Here DCD is approximately 60 times faster than DCR in obtaining the results.

The results of Simulation 3 are shown in Fig. 2.10. Here there exist three true change points, the first at time 125, the second at time 500, and the third at time 750. Clearly, both algorithms do an excellent job of detecting the true change points. Here DCD is approximately 30 times faster than DCR in obtaining the results.

Fig. 2.11 shows the results of Simulation 4. Again, both algorithms do an excellent job of detecting the true change point, which is located at time 100, but DCD does so with a 20-fold increase in speed.

Finally, the results of Simulations 5 and 6 are shown in Figs. 2.12 and 2.13, respectively. In both cases the algorithms do an excellent job of detecting the true

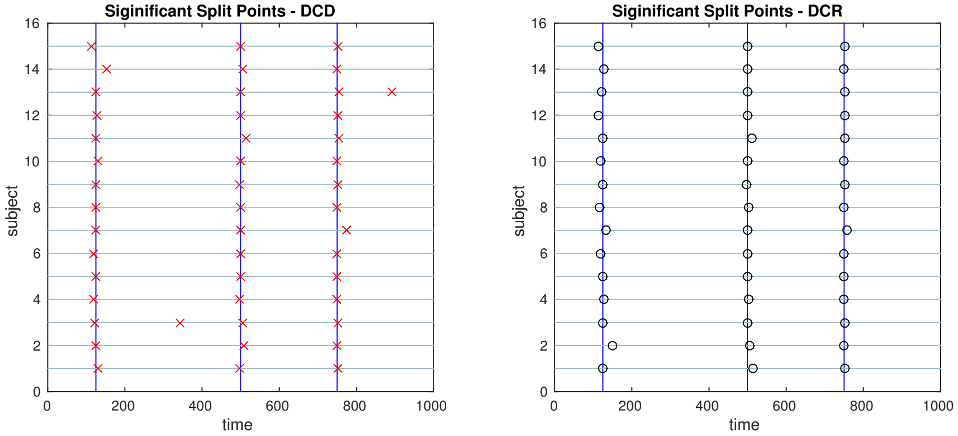

Figure 2.10: The results of Simulation 3. (Left) The red crosses show significant split points found by DCD. (Right) The blue circles show significant split points found by DCR. The blue vertical lines indicate the timing of the true change points.

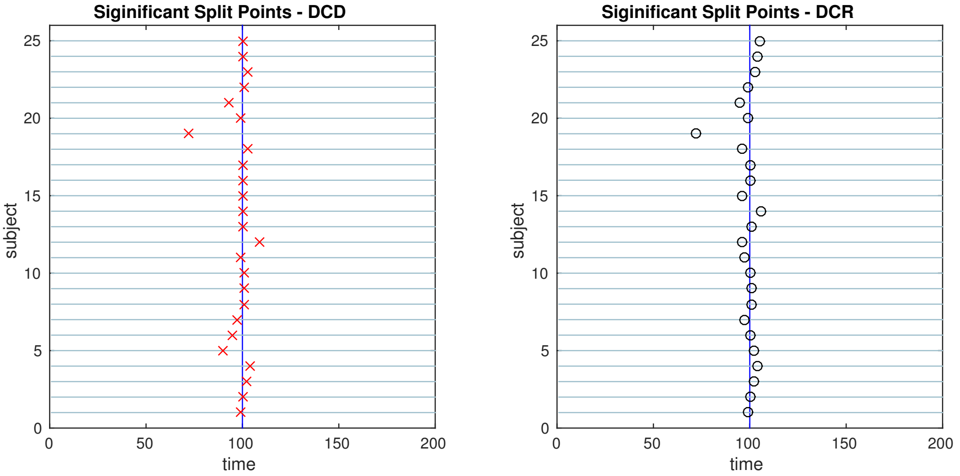

Figure 2.11: The results of Simulation 4. (Left) The red crosses show significant split points found by DCD. (Right) The blue circles show significant split points found by DCR. The blue vertical line indicates the true change points.

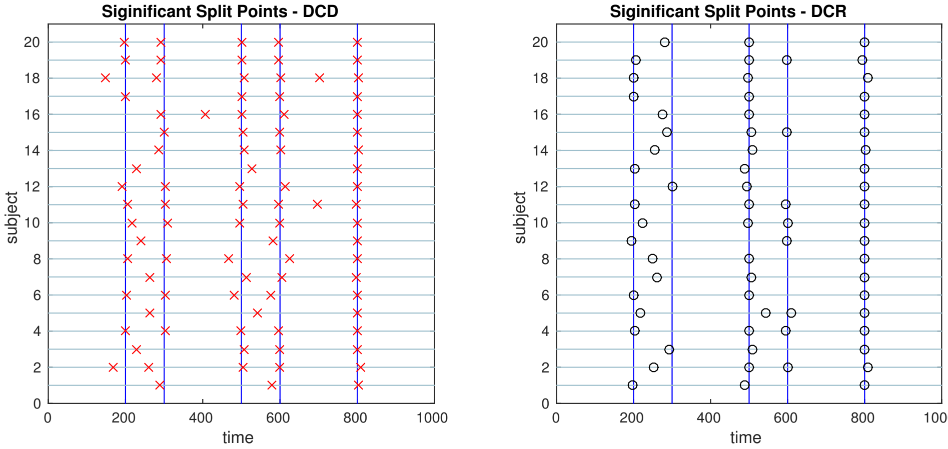

Figure 2.12: The results of Simulation 5. (Left) The red crosses show significant split points found by DCD. (Right) The blue circles show significant split points found by DCR. The blue vertical lines indicate the timing of the true change points.

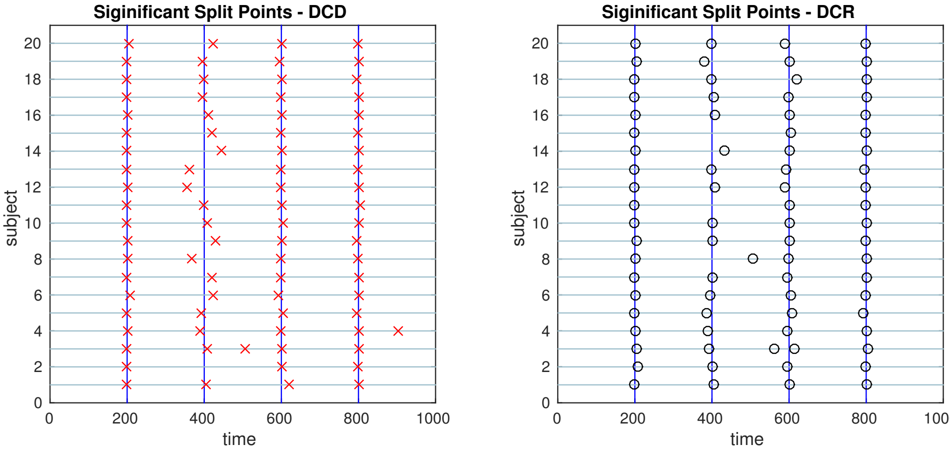

Figure 2.13: The results of Simulation 6. (Left) The red crosses show significant split points found by DCD. (Right) The blue circles show significant split points found by DCR. The blue vertical lines indicate the timing of the true change points.

Simulation	1	2	3	4	5	6
Correct Zero Rate	0.9497	0.9568	0.9474	0.9529	0.4955	0.9461
Correct Non-zero Rate(TP)	1	0.9763	0.9644	0.6535	0.9837	0.9628
False Positive Rate (average)	0.0503	0.0432	0.0526	0.0471	0.5045	0.0539

Table 2.2: Sparsity control results of covariance matrices

change points. However, DCD does so with a 30-fold increase in speed in both cases.

Although the main goal of DCD is to detect change points, and the estimation of a connectivity graph seems a byproduct, the accuracy of the covariance matrix or precision matrix estimation leads to better change point detection, and vice versa. Using the Adaptive Thresholding Approach, we need to control the family-wise error rate or false discovery rate. The estimation of a J-dimensional covariance matrix requires $O(J^2)$ hypothesis tests. In our simulation examples, we adjust the significance level η by η/J, to guard against being as conservative as Bonferroni correction, while still obtaining adequate control over the family-wise error rate. Results show that the estimation of the sparsity structure is accurate in most simulations. The list of the average proportion of correctly identified zero/non-zero elements of the covariance matrices are listed in Table 2.2.

In summary, in each of the “low dimensional” simulations described above, with the number of ROIs ~ 20, DCR achieves similar results as DCD with a significant speed-up in runtime. However, to investigate how well the methods scale to a more “high dimensional” settings, we expand upon two of the simulations to inspect how computational time changes as a function of the number of ROIs for the two algorithms.

In the first (denoted 2B), we generated 80 ROIs data for 50 subjects under the same settings as described in Simulation 2. Here only the first 20 nodes contain information, and the remaining are simply white noise. We ran DCD and DCR using ROIs 1:r, where r ranged from 20 to 80 in increments of 5. In the second (denoted 4B), we generated 70 ROIs for 50 subjects under the same settings as described in Simulation 4. Here only the first 5 nodes contain information, while all remaining nodes are white noise. We ran DCD and DCR on a subset of ROIs numbered 1:r, where r ranged from 5 to 70 in increments of 5.

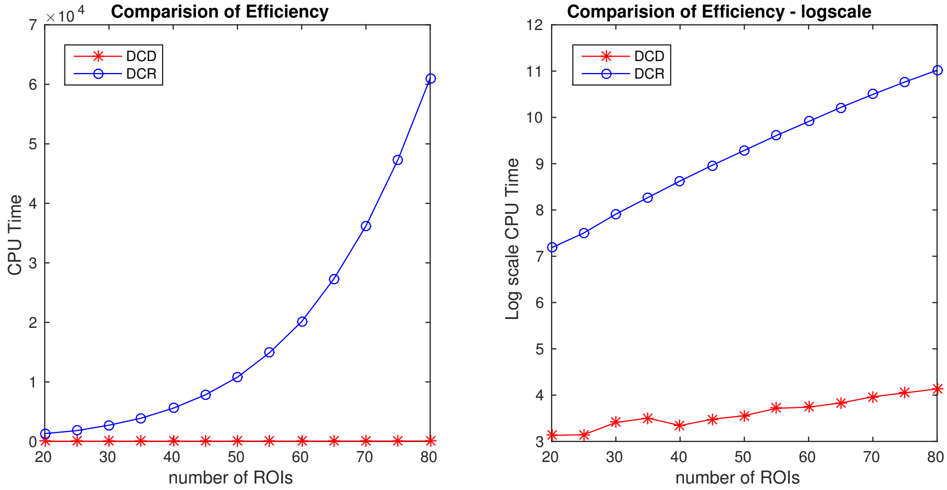

Figure 2.14: Runtime for Simulation 2B as a function of number of nodes for both DCD and DCR on both regular (left) and log-scale (right). Clearly, DCD scales much better than DCR

The results of Simulation 2B are summarized in Figs. 2.14 and 2.15. From Fig. 2.14 it is clear that the computation time for DCR increases exponentially

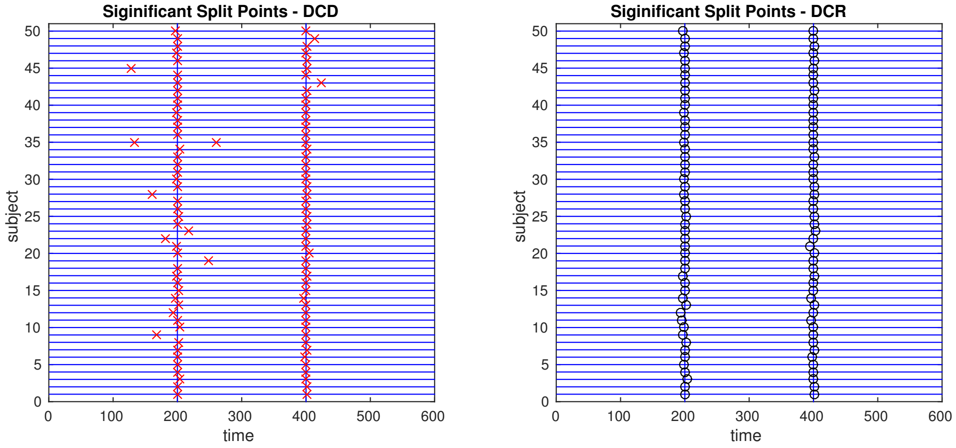

Figure 2.15: The results of Simulation 2B. (Left) The red crosses show significant split points found by DCD. (Right) The blue circles show significant split points found by DCR. The blue vertical lines indicate the timing of the true change points.

with the number of ROIs, while the computation time for DCD is much shorter and nearly linear. Though the results of DCR appear slightly better than DCD (see Fig. 2.15), with less deviations from the true change points, this comes at a substantial computational cost.

The results of Simulation 4B are summarized in Figs. 2.16 and 2.17. Based on Fig. 2.16 it is clear that the computation time for DCR increases exponentially with the number of ROIs, while the computation time for DCD is much shorter and has a near linear increase. In addition, judging by Fig. 2.17 the algorithm also appears to more accurately detect the timing of the true change points.

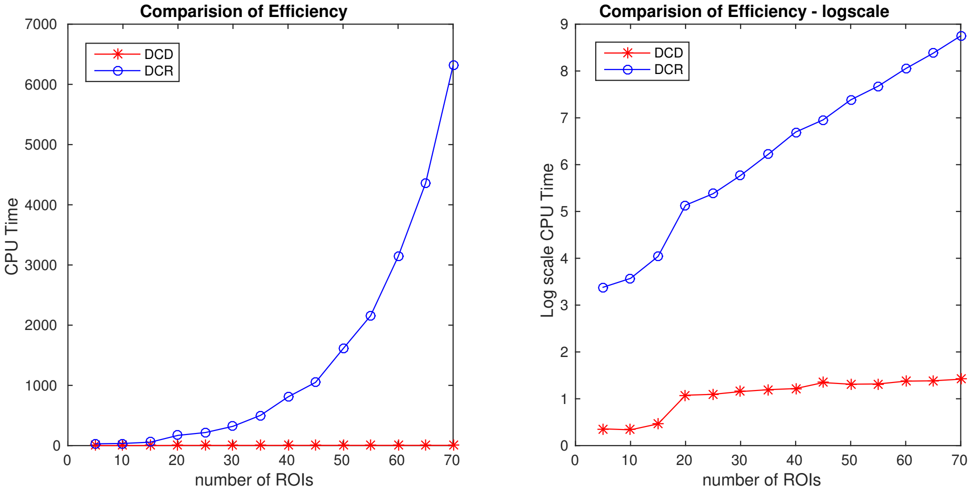

Figure 2.16: Runtime for Simulation 4B as a function of number of nodes for both DCD and DCR on both regular (left) and log-scale (right). Clearly, DCD scales much better than DCR.

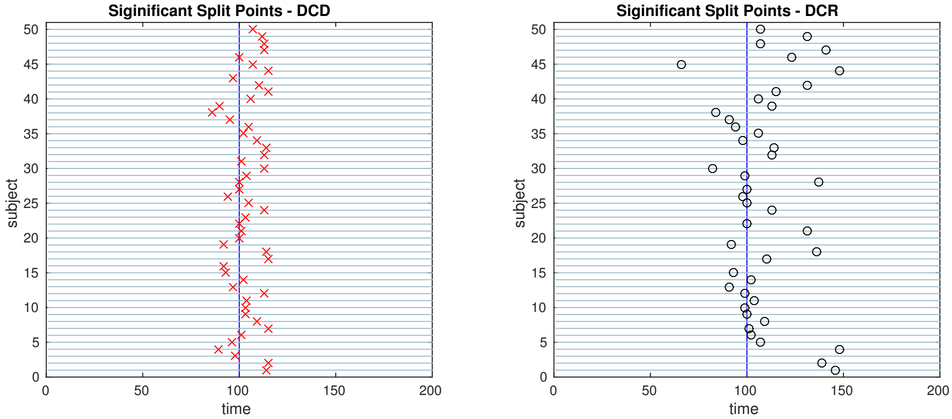

Figure 2.17: The results of Simulation 4B. (Left) The red crosses show significant split points found by DCD. (Right) The blue circles show significant split points found by DCR. The blue vertical line indicates the true change point.

2.4 Application to Experimental Data

2.4.1 Social Evaluative Threat Experiment

The data was taken from an experiment where subjects performed an anxiety-inducing task while fMRI data was acquired (Wager et al., 2009). This is the same data set used in the previous DCR papers (Cribben et al., 2012, 2013), as well as in other papers exploring mean change points (Lindquist et al., 2007; Robinson et al., 2010). The task was a variant of a well-studied laboratory paradigm for eliciting social threat, during which participants were asked to give a speech under evaluative pressure. It consisted of an off-on-off design, with an anxiety-provoking speech preparation task sandwiched between two lower-anxiety rest periods. Prior to the scanning session, subjects were informed that they were to be given 2 min to prepare a 7 min speech, the topic of which would be revealed to them during scanning, that would be delivered to a panel of expert judges after the scanning session. However, they were told that there was a small chance that they would be randomly selected not to give the speech. After the start of fMRI acquisition, during the initial 2 min resting period subjects viewed a fixation cross. At the end of this period, an instruction slide appeared describing the speech topic for 15 s ('why you are a good friend'). The slide instructed subjects to prepare enough for the entire 7 min period. After 2 min of silent preparation, a second instruction screen appeared for 15 s that informed subjects that they would not have to give the speech. The functional run concluded with an additional 2 min period of resting baseline.

During the course of the experiment a series of 215 functional images were acquired (TR = 2 s). A detailed description of the data acquisition and preprocessing can be found in previous work (Wager et al., 2009). In order to create ROIs, time series of voxels were averaged across pre-specified regions of interest. We used data consisting of 4 ROIs and heart rate for 23 subjects. The 4 ROIs were chosen due to the fact that they showed a significant relationship to heart rate in an independent data set. They included the ventral medial prefrontal cortex (VMPFC), the anterior medial prefrontal cortex (mPFC), the striatum/pallidum, and the dorsal lateral prefrontal cortex (DLPFC)/inferior frontal junction (IFJ). The temporal resolution of the heart rate was 1 s compared to 2 s for fMRI data, so it was down-sampled by taking every other measurement.

Both the DCD and DCR approaches were applied to the 23 subjects individually. For the DCD algorithm, we used $(\alpha, \beta, \eta) = (0.1, 0.1, 0.05)$ as input parameters, and the runtime was 0.92 s. For the DCR algorithm, we adopted similar parameter settings used in (Cribben et al., 2013), where we used the following settings: $\Delta = 40$, $\lambda - list = (1, 2^{-1}, ..., 2^{-9})$, $\alpha = 0.1$, $N_b = 50$ and $\xi = 20$. The runtime for DCR was 32.14 s.

The change points detected by the two algorithms are displayed in Fig. 2.18. Both consistently give rise to change points around the time of the first visual cue. In addition, there appear to be changes towards the middle of speech preparation

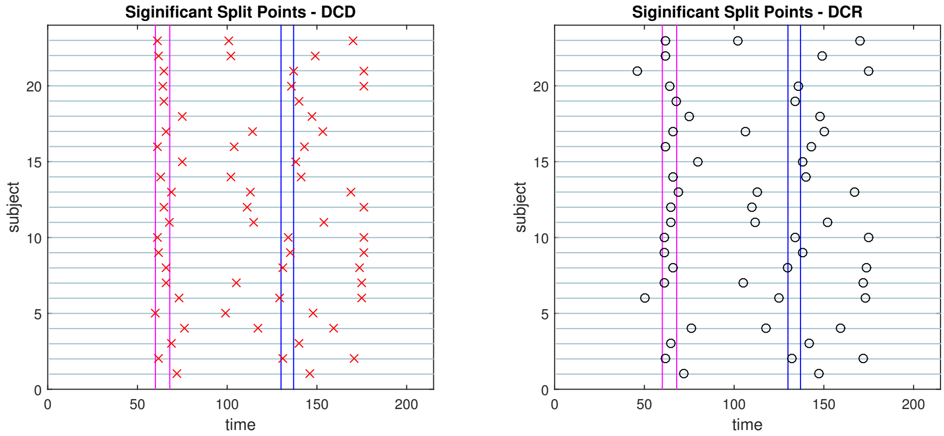

Figure 2.18: Results of the social evaluative threat experiment, with data consisting of four ROIs and heart rate. The x-axis represents time and y-axis depicts the subject number. The vertical lines represent the timing of the instruction slides. (Left) Red crosses show the change points identified by DCD. (Right) The black circles show the change points obtained via DCR.

and around the time of the second visual cue, though these are less consistent across subjects. Interestingly, in contrast to the DCR algorithm, the first change points given by the DCD algorithm appears to coincide more closely to the timing of the first introduction cue. Otherwise the number, and placement, of the detected change points are roughly equivalent across methods.

2.4.2 Human Connectome Project

To study DCD's performance on high dimensional data, we applied the method to resting-state fMRI (rfMRI) data from the 2014 Human Connectome Project (HCP) data release (Van Essen et al., 2013). The data consists of 4 separate 15 min rfMRI runs, each consisting of 1200 time points, collected for each of 468 subjects. Each run was minimally preprocessed according to the procedure outlined in (Glasser et al., 2013), with artifacts removed using FIX (FMRIB's ICA-based Xnoiseifier) (Griffanti et al., 2014; Salimi-Khorshidi et al., 2014). Each data set was temporally demeaned with variance normalization applied according to (Beckmann and Smith, 2004). Group-PCA output was generated by applying MELODICs Incremental Group-PCA on the 468 subjects. This comprises the top 4500 weighted spatial eigenvectors from a group-averaged PCA. The output was fed into group-ICA using FSL's MELODIC tool (Beckmann and Smith, 2004), applying spatial-ICA with 100 distinct ICA components. The set of ICA spatial maps were mapped onto each subject's time series data to obtain a single representative time series per ICA component using the 'dual-regression' approach, in which the full set of ICA maps are used as spatial regressors against the full data (Filippini et al., 2009).

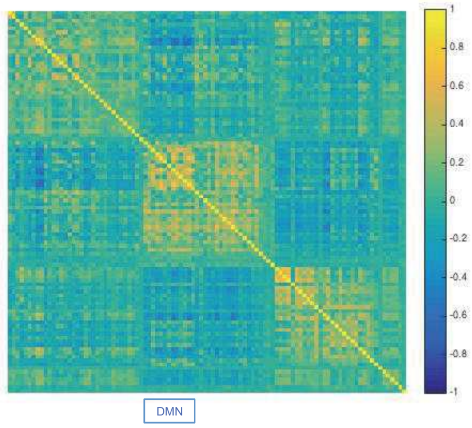

Figure 2.19: Results of the analysis of the HCP data. The static correlation matrix for a single subject (100307), computed using data from the four runs. Components corresponding to the default mode network are highlighted by DMN.

For illustration purposes we applied DCD to data consisting of 100 ICA component time courses from a single subject (100307). We began by computing the static correlation matrix for the subject by concatenating data across the four runs. The resulting correlation matrix was sorted using the Louvain algorithm (Blondel et al., 2008), which has proven efficient at identifying communities in large networks. The resulting correlation matrix can be seen in Fig. 2.19. There are clear groupings of similar components that correspond to common networks seen in the resting-state literature, including the visual, somatomotor, cognitive control, and default mode networks.

Next, we applied DCD with input parameters $(\alpha, \beta, \eta) = (0.05, 0.05, 0.02)$ to each of the four runs. The runtime for each was less than 10 seconds. The correlation matrices for all partitions are displayed in Fig. 2.20, along with the corresponding temporal partition listed above them. Each run consisted of either 6 or 7 partitions, and there are clear similarities in connectivity states between runs. Here one would not expect the timing of the change points to be similar across runs, as there is no explicit task designed to invoke state changes. Rather, this example is primarily meant to illustrate that DCD is able to detect change points in situations where there are 100 nodes.

That said, these results are consistent with results seen in previous literature (Allen et al., 2012), and suggest that dynamic behavior of functional connectivity is present in the resting state data. In particular states appear to be differentiated by

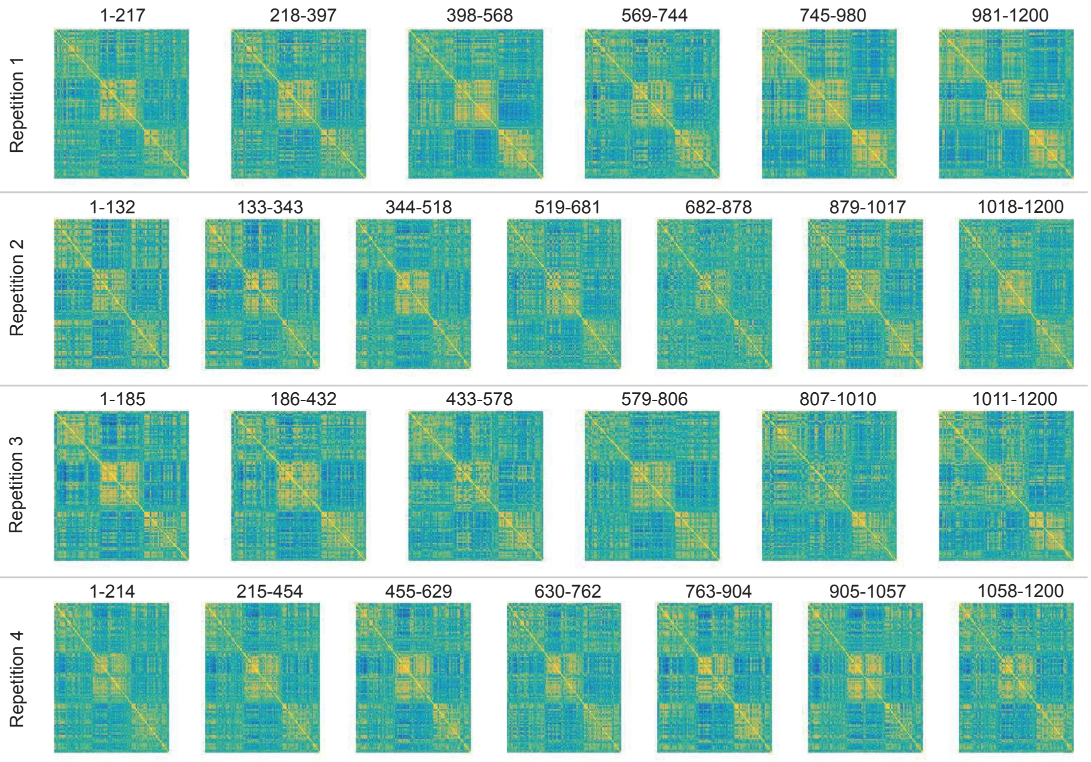

Figure 2.20: Results of the analysis of the HCP data using DCD. Each row depicts the estimated correlation matrices for the time partitions detected by DCD for each of 4 runs for subject 100307. Above each matrix is the temporal information for each time partition.

connectivity between default mode components, and between default mode components and other components throughout the brain.

2.5 Discussion

In this work, we have developed a novel algorithm for change point detection in fMRI data. It partitions the fMRI time series into sequences based upon functional connectivity changes between ROIs or voxels, as well as mean activation changes. DCD can be applied to time series data from ROI studies, or to temporal components obtained from either a principal components or independent components analysis. Its data-driven design means it does not require any prior knowledge of the nature of the experiment. In addition, the accuracy of the result on single subject data allows for analysis on experiments where one expects large heterogeneity in connectivity across subjects and between runs, such as in resting state fMRI data.

To reduce the burden on users, all three input parameters to the DCD algorithm have a clear statistical interpretation, making it easy to use even for those unfamiliar with the intrinsic details of the algorithm. As long as the user has a basic understanding of hypothesis testing, they should have the appropriate knowledge necessary to alter the parameters in order to improve the performance of the algorithm.

We contrast the approach to the previously introduced DCR technique, which also seeks to find connectivity change points. The most significant advantage of

DCD compared to DCR is its computational efficiency, driven in large part by the newly proposed adaptive thresholding schema for sparse covariance matrix estimation. Based on the results of two high-dimensional simulation studies, as well as further empirical studies, we found that the computation time for DCR grows rapidly with an increased number of ROIs. Thus, when the number of regions exceeds 50, the computational burden of DCR can be intimidating for most users. In contrast, the computation time of DCD increases roughly linearly, and can easily handle hundreds of ROIs, in a matter of minutes for most general fMRI experimental settings.

In the DCD algorithm, we choose to maximize the total likelihood function instead of the Bayesian information criterion (BIC) that is used in the DCR algorithm. The design of the DCD algorithm frees the user from performing model selection from a list of regularization parameters, so that we can use the likelihood function as a more natural criterion. Furthermore, utilizing the likelihood function avoids a common problem arising when applying the BIC; namely that when adding the BIC score of two subsets of lengths n_1 and n_2 ($n_1 + n_2 = n$), consisting of roughly the same number of parameters $k_1 \approx k_2 \approx k$, the total penalty term is $klog(n_1) + klog(n_2) \propto log\big(n_1(n - n_1)\big)$, which tends to favor small or large n_1 when minimizing the BIC. This is the reason for the apparent cluster of false positives obtained using DCR at time points Δ and $T - \Delta$, shown in Fig. 2.8.

Another critical difference between the two algorithms is the manner in which sparsity is enforced. DCR uses GLASSO, and thus places sparsity constraints on the precision matrix, while DCDs adaptive thresholding approach places them on the covariance matrix. The former may be more natural in the fMRI setting, due to the relationship between the precision matrix and the connectivity graph where zero elements correspond to conditional independence. However, we found in our simulation studies that when estimating connectivity change points it does not appear to be critical upon which matrix we impose sparsity, and the computational advantages of operating on the covariance matrix becomes increasingly attractive. However, in settings where the precision matrix is sparse, and the corresponding covariance matrix is dense, DCD can potentially run into problems and alternative approaches should be explored.

One limitation preventing us from further improving the runtime of the DCD algorithm comes from the nature of greedy method we used for maximizing the likelihood. The greedy search strategy makes the locally optimal choice at each step, but cannot ensure the global optimum solution is obtained. However, as a data-driven method, the results from DCD will still provide a reasonable starting point for exploring the experimental data. Another disadvantage of DCD are limits on the types of experiments it may be applied to. In this work, we have demonstrated its effectiveness using both blocked-design task fMRI experiments as well as resting state data. However, for event-related designs, the brain connectivity and activity level may change too rapidly to be able to obtain a valid estimate from DCD. Hence, when the DCD algorithm detects no significant change points, it may in fact be the case that the activity pattern changes too frequently to be detected.

Similar to group-level DCR, there is also a simple variant of DCD for group inference, which stacks subjects and calculates the summation of the likelihood function in each step. This approach can be used in experiments where one expects subjects to change states at similar time points (e.g., in the social evaluative threat experiment), and is not recommended for resting-state experiments where subjects are not expected to behave in a similar manner. In general, we suggest one first performs single-subject DCD, and if the resulting change points show synchronization across a subset of subjects, then apply group-level DCD to obtain more accurate results. Due to the flexibility of the DCD algorithm, we can also incorporate the GLASSO technique for sparse precision matrix estimation in place of adaptive thresholding method, which may also lead to improved accuracy at the cost of slower runtime.

In sum, the newly proposed DCD algorithm is a fast and efficient approach towards detecting changes in functional connectivity, especially for experiments where the nature, timing or duration of the involved psychological processes are unknown.

Chapter 3

Evaluating Dynamic Bivariate Correlations in Resting-state fMRI: A comparison study and a new approach

CHAPTER 3. DYNAMIC BIVARIATE CORRELATIONS IN RS-FMRI

Abstract

To date, most functional Magnetic Resonance Imaging (fMRI) studies have assumed that the functional connectivity (FC) between time series from distinct brain regions is constant across time. However, recently, there has been increased interest in quantifying possible dynamic changes in FC during fMRI experiments, as it is thought this may provide insight into the fundamental workings of brain networks. In this work we focus on the specific problem of estimating the dynamic behavior of pair-wise correlations between time courses extracted from two different regions of the brain. We critique the commonly used sliding-windows technique, and discuss some alternative methods used to model volatility in the finance literature that could also prove useful in the neuroimaging setting. In particular, we focus on the Dynamic Conditional Correlation (DCC) model, which provides a model-based approach towards estimating dynamic correlations. We investigate the properties of several techniques in a series of simulation studies and find that DCC achieves the best overall balance between sensitivity and specificity in detecting dynamic changes in correlations. We also investigate its scalability beyond the bivariate case to demonstrate its utility for studying dynamic correlations between more than two brain regions. Finally, we illustrate its performance in an application to test-retest resting state fMRI data.

KEYWORDS: Functional connectivity, fMRI, Dynamics, Resting state, Dynamic conditional correlations

3.1 Introduction

To date, most functional Magnetic Resonance Imaging (fMRI) studies have implicitly assumed that the functional connectivity (FC) between time series from distinct brain regions is constant across time. Recently, there has been an increased interest in attempting to quantify the dynamic changes in FC during the course of an fMRI experiment (e.g., (Allen et al., 2012; Chang and Glover, 2010; Handwerker et al., 2012; Hutchison et al., 2013; Jones et al., 2012; Kiviniemi et al., 2011; Cribben et al., 2012)); particularly during resting state. Changes in both the strength and directionality of functional connections have been observed to vary across experimental runs (Hutchison et al., 2013). It is thought that this temporally varying information may be used to provide insight into the fundamental properties of brain networks. For example, studies have found correlation between changes in resting-state FC and simultaneous recorded electrophysiological data, as well as behavioral data (Allen et al., 2012; Chang et al., 2013; Tagliazucchi et al., 2012; Thompson et al., 2012), indicating a possible neuronal origin to the observed variation.

Though it is of increasing importance, interpreting temporal fluctuations in FC can be difficult due to low signal-to-noise ratio, physiological artifacts and variation in BOLD signal mean and variance over time (Hutchison et al., 2013). For these reasons, it is often difficult to determine whether observed fluctuations in FC should be attributed to neuronal activity or whether they are simply due to random noise, and thus significant research is still needed in the area. In particular, there remains uncertainty regarding the appropriate analysis strategy to use and how to properly interpret results.

While numerous metrics for evaluating functional connectivity exist, in this work we focus exclusively on pair-wise correlations between time courses from two regions of the brain. The most common approach towards studying these types of dynamic correlations has been the so-called sliding-window approach (Allen et al., 2012; Chang and Glover, 2010; Handwerker et al., 2012). Here, a window of fixed length w is selected, and data within that window are used to compute the correlation coefficient. The window is thereafter moved step-wise across time, providing a time-varying measure of the correlation between brain regions.

Though extremely simple to implement, this approach has a number of potential shortcomings, including the use of arbitrary window lengths, an inability to deal with abrupt changes and the fact that it gives equal weight to all observations that lie less than w time points in the past, but no weight to older observations (Engle, 2002a; Lebo and Box-Steffensmeier, 2008). In this work we evaluate the properties of the sliding-window technique, as well as introduce several alternative model-based approaches that we believe may ultimately prove more useful for analyzing dynamic correlations in fMRI.

The problem of studying time-varying variances and correlations between multivariate time series, while relatively recent in the neuroimaging literature, has been extensively studied in the finance literature during the past few decades (see, for example, (Bauwens et al., 2006) for a nice overview). Financial time series often exhibit time-varying conditional standard deviations (typically referred to as volatility) and correlations. Here volatility is seen as a measure of the risk of certain assets, while correlations between time series play an important role in asset allocation, risk management and portfolio selection (Tsay, 2006).

In the finance literature, simple methods such as sliding-windows (often referred to as "rolling-windows") and related techniques have been widely used to model dynamic correlations. However, it is generally accepted that time series models provide superior results (Hansen and Lunde, 2005) compared to rolling-window methods. In particular, generalized autoregressive conditional heteroscedastic (GARCH) models (Bollerslev, 1986) have been shown to efficiently model both dynamic variances and correlations. These techniques model covariance in an analogous manner as fMRI noise is modeled in standard GLM analysis using more conventional time series analysis techniques such as autoregressive (AR) and autoregressive moving-average (ARMA) models (Purdon et al., 2001; Lindquist, 2008). Recently, (Engle, 2002a) introduced a particular variant of the multivariate GARCH model, denoted the dynamic conditional correlation (DCC) model, which has been shown to be particularly effective for estimating both time-varying variances and correlations (Lebo and Box-Steffensmeier, 2008). In this approach all the parameters are effectively estimated through quasi-maximum likelihood methods and require no *ad hoc* parameter settings. The model uses a sequential estimation scheme and a parsimonious parameterization that allows it to estimate potentially large covariance structures. Hence, the method promises to be scalable to situations where one wants to study dynamic correlations between more than two brain regions.

It is our belief that lessons from finance may help inform future studies of dynamic correlation in the neuroimaging context as well. To this end, in this work we introduce multivariate volatility models to the neuroimaging literature. Further, we investigate the properties of these and other commonly used techniques in a series of simulation studies. Our results indicate that techniques such as DCC show great promise in the analysis of dynamic FC. We conclude by applying the method to test-retest resting state fMRI data.

3.2 Methods

In this section we begin by setting up the problem and continue by introducing several methods for estimating dynamic correlations, which we seek to compare in our simulation studies. These include both sliding-window type methods and model-based multivariate volatility methods.

3.2.1 Problem Set-up

Let us assume that we are interested in studying the relationship between two time series $y_{1,t}$ and $y_{2,t}$, measured over two separate regions of interest (ROIs) in the brain, at equally spaced time points $t = 0, \ldots T$. Let $\mathbf{y}_t = (y_{1,t},\, y_{2,t})^T$ be a vector containing the values of both time series at time t and assume that

$$\mathbf{y}_t = \mu_t + \mathbf{e}_t, \tag{3.1}$$

where $\mu_t = (\mu_{1,t},\, \mu_{2,t})^T$ is the conditional mean of $\mathbf{y}_t$ using all information in the time series observed up to time t, denoted $E_{t-1}(\mathbf{y}_t)$. The noise term $\mathbf{e}_t$ has mean zero and its conditional covariance matrix at time t can be expressed as

$$\Sigma_t = \begin{pmatrix} \sigma^2_{1,t} & \sigma_{12,t} \\ \sigma_{12,t} & \sigma^2_{2,t} \end{pmatrix}. \tag{3.2}$$

Here the diagonal terms represent the conditional variance of $y_{i,t}$ using all information in the time course observed up to time t, written $\sigma^2_{i,t} = E_{t-1}((y_{i,t} - \mu_{i,t})^2)$ for $i = 1, 2$. The square root of this entity, σ_{it}, is typically referred to as the volatility of the time series in the financal literature. The off-diagonal term is $\sigma_{12,t} = \sigma_{1,t}\sigma_{2,t}\rho_t$ where

$$\rho_t = \frac{E_{t-1}((y_{1,t} - \mu_{1,t})(y_{2,t} - \mu_{2,t}))}{\sqrt{E_{t-1}((y_{1,t} - \mu_{1,t})^2)E_{t-1}((y_{2,t} - \mu_{2,t})^2)}} \tag{3.3}$$

represents the conditional correlation coefficient. Under this definition the conditional correlation at time t relies on information that is observed up to time $t-1$. Importantly, this quantity is guaranteed to lie in the interval $[-1, 1]$ for all possible realizations of these random variables, as well as, for any linear combination.

Throughout we assume without loss of generality that $\mu_t = \mathbf{0}$ and $\mathbf{y}_t = \mathbf{e}_t$. Under this assumption, Eq. 3.3 simplifies as follows:

$$\rho_t = \frac{E_{t-1}(y_{1,t}y_{2,t})}{\sqrt{E_{t-1}(y_{1,t}^2)E_{t-1}(y_{2,t}^2)}}. \tag{3.4}$$

The conditional covariance matrix defined in Eq. 3.2 can alternatively be written in matrix form as

$$\Sigma_t = D_t R_t D_t \tag{3.5}$$

where D_t is a diagonal matrix consisting of the conditional standard deviations of the time series, i.e. $D_t = diag\{\sigma_{1,t},\ \sigma_{2,t}\}$ and R_t is the correlation matrix,

$$R_t = \begin{pmatrix} 1 & \rho_t \\ \rho_t & 1 \end{pmatrix}. \tag{3.6}$$

Throughout this paper, our primary concern will be in developing methods for estimating the components of the conditional covariance matrix, with the particular goal of estimating $\sigma_{1,t}^2$, $\sigma_{2,t}^2$ and ρ_t. We begin by discussing sliding-windows techniques and move on to discuss multivariate volatility models.

3.2.2 Sliding-Window Techniques

Perhaps the simplest approach towards estimating the elements of the covariance matrix is to use the sliding-window technique. In particular, sliding-window correlations have received substantial interest in the recent neuroimaging literature (Allen et al., 2012; Chang and Glover, 2010; Handwerker et al., 2012). Here, a time window of fixed length w is selected, and data points within that window are used to calculate the correlation coefficients. The window is thereafter moved across time and a new correlation coefficient is computed for each time point.

In recent work, (Chang and Glover, 2010) define the sliding-window correlation at time t as follows:

$$\hat{\rho}_t = \frac{\sum_{s=t}^{t+w-1} y_{1,s} y_{2,s}}{\sqrt{(\sum_{s=t}^{t+w-1} y_{1,s}^2)(\sum_{s=t}^{t+w-1} y_{2,s}^2)}}. \tag{3.7}$$

According to this definition the correlation at time t is based on w future measurements of the time courses. Though it should be noted that in resting state experiments, which is the focus of their work, the exact timing of the dynamic correlation is not meaningful. We still prefer to define the window using only past values as it provides a more suitable estimate of the conditional correlation defined in Eq. 3.4.

For this reason, we define the sliding-window so that it gives equal weight to all observations that lie less than w time points in the past and zero weight to all older observations. Hence, the general form of the estimate is given by

$$\hat{\rho}_t = \frac{\sum_{s=t-w-1}^{t-1}(y_{1,s} - \widehat{\mu}_{1,s})(y_{2,s} - \widehat{\mu}_{2,s})}{\sqrt{(\sum_{s=t-w-1}^{t-1}(y_{1,s} - \widehat{\mu}_{1,s})^2)(\sum_{s=t-w-1}^{t-1}(y_{2,s} - \widehat{\mu}_{1,s})^2)}}, \tag{3.8}$$

where $\widehat{\mu}_{i,t}$, $i = 1, 2$ represents the estimated time-varying mean. Assuming $\mu = 0$, we can instead use

$$\hat{\rho}_t = \frac{\sum_{s=t-w-1}^{t-1} y_{1,s}y_{2,s}}{\sqrt{(\sum_{s=t-w-1}^{t-1} y_{1,s}^2)(\sum_{s=t-w-1}^{t-1} y_{2,s}^2)}}. \tag{3.9}$$

Though not commonly performed in the neuroimaging literature, it is a simple extension to use the sliding-window technique to estimate the conditional variance. We define, for the case when $\mu = 0$, the sliding-window variance at time t as follows:

$$\hat{\sigma}_{i,t}^2 = \frac{1}{w-1} \sum_{s=t-w-1}^{t-1} y_{i,s}^2. \tag{3.10}$$

While the sliding-window technique allows for a simple approach for exploring changes in connectivity, it has a number of obvious shortcomings. First, it gives equal weight to all observations less than w time points in the past and 0 weight to all others. Hence, the removal of a highly influential outlying data point will cause a sudden change in the dynamic correlation that may be mistaken for an important aspect of brain connectivity. To circumvent this problem, (Allen et al., 2012) suggested the use of a tapered sliding-window. Here, the sliding-window (width = 22 TRs) is convolved with a Gaussian kernel ($\sigma = 3$ TRs). This allows points to gradually enter and exit from the window as it moves across time. It should be noted that they define t to be the middle of the subsequent window, thus giving equal weight to future and past values.

Second, the window length is typically chosen in an arbitrary manner. While methods exist for automatically selecting window lengths (see for example (Ombao and Van Bellegem, 2008) which proposes a method for automatically selecting the optimal window width for estimating local coherence), these methods have yet to find wide usage in the field. While it is beneficial to have a short window to better detect transient changes in connectivity, a large window is often necessary to allow for robust estimation of the correlation coefficient. Fig. 3.1 illustrates the effect of window length on the estimated conditional correlation fit to null data (i.e. two uncorrelated time courses). Note, that when using short windows, correlations are susceptible to large variations. In fact, a 95% confidence interval for null data will lie roughly between $\pm 2/\sqrt{w}$, which for a window length of 15 observations would roughly lie between $[-0.5, 0.5]$. Hence, using these settings random fluctuations can mistakenly be seen as meaningful time-varying correlations. The width of the 95% confidence interval is halved if we increase the window length four-fold to 60 observations. However, this comes at the cost of reduced sensitivity to minor changes in correlation.

3.2.3 Multivariate Volatility Models

In this section we discuss two multivariate volatility models commonly used in the finance literature. The first, denoted the exponential weighted moving average (EWMA) approach, shares some similarities to sliding-windows. However, it provides solutions to some of the more obvious shortcomings of that approach. The second, the DCC model, provides a parametric approach towards estimating dynamic correlations, much like auto-regressive (AR) and auto-regressive moving averages (ARMA) models provide a parametric approach towards modeling fMRI noise (Purdon et al., 2001).

3.2.3.1 EWMA

The first multivariate volatility model we introduce provides an alternative to the tapered sliding-window. The EWMA approach applies declining weights to past observations in the time series based on a parameter λ, and is based upon the following recursion

$$\Sigma_t = (1-\lambda)\mathbf{e}_{t-1}\mathbf{e}'_{t-1} + \lambda\Sigma_{t-1}, \tag{3.11}$$

where Σ_t is the conditional covariance matrix. This approach places the most weight on recent observations, and for each step away from t values become gradually downweighted by a factor λ, before eventually being removed from further computations.

Decomposing the covariance matrix in Eq. 3.11, we can express the conditional variances and covariance as follows:

$$\begin{aligned}\sigma^2_{i,t} &= (1-\lambda)e_{i,t-1}e'_{i,t-1} + \lambda\sigma^2_{t-1} \\ &= (1-\lambda)y_{i,t-1}y'_{i,t-1} + \lambda\sigma^2_{t-1} \end{aligned} \tag{3.12}$$

and

$$\sigma_{12,t} = (1-\lambda)e_{1,t-1}e'_{2,t-1} + \lambda\sigma_{12,t-1}. \tag{3.13}$$

The parameter λ must take values between 0 and 1, and it determines how responsive the estimate of the covariance matrix is to the most recent time points. A small value of λ gives high weight to recent time points, while a large value produces estimates that respond more gradually to new information. The value determines how many data points are included in the calculation and serves the equivalent purpose to the window size used in the sliding-window approach. Another important property of the approach is that as long as the recursion is initialized with a positive-definite matrix, it will remain so throughout the sequence.

Often the value of λ is set arbitrarily, with 0.94 being a popular value in the finance literature (Sheppard, 2012). However, if one assumes $\mathbf{y}_t$ to be bivariate Gaussian, it is straightforward to estimate λ through maximum likelihood estimation. Here, the log-likelihood function that should be maximized can be written:

$$\log\{L(\lambda)\} \propto -\frac{1}{2}\sum_{t=1}^{T}|\Sigma_t| - \frac{1}{2}\sum_{t=1}^{T}\mathbf{e}_t'\Sigma_t^{-1}\mathbf{e}_t. \tag{3.14}$$

This function can be maximized using any standard search algorithm and the value that λ takes at the optimum recorded.

Fig. 3.2 illustrates the effect of λ on the estimated conditional correlation fit to null data (i.e. two uncorrelated time courses). Note, that as λ gets smaller, less of the past data is used in the calculation of the correlation. Hence, if one uses small values of λ one is susceptible to similar problems that arise when using short window lengths when computing sliding-window correlations. Hence, it is beneficial to estimate this important parameter directly from the data.

Finally, it should be noted that a related EWMA approach has previously been used to detect dynamic changes in the mean BOLD response (Lindquist et al., 2007). There the goal was to detect so-called “change points” representing state-changes where the mean signal changes values, thus allowing one to model slowly varying processes with uncertain onset times and durations of underlying psychological activity. In contrast, this work considers dynamic correlations, which are perhaps more relevant for resting state analyses.

3.2.3.2 DCC

The second multivariate volatility model we introduce is the DCC model (Engle, 2002a), which is an approach towards estimating the conditional variances and correlations that has become increasingly popular in the finance literature during the past decade. However, before introducing DCC, we must first discuss GARCH processes (Engle, 1982; Bollerslev, 1986), which are often used to model volatility in univariate time series. They provide flexible models for the variance in much the same manner that commonly used time series models, such as ARMA and AR, provide models for the mean. GARCH models express the conditional variance of a single time series at time t as a linear combination of past values of the conditional variance and of the squared process itself. To illustrate, let us assume that we are observing a univariate process

$$y_t = \sigma_t \epsilon_t \tag{3.15}$$

where ϵ_t is a $N(0,1)$ random variable and σ_t represents the time-varying variance term we seek to model. In a GARCH(1,1) process the conditional variance is expressed as

$$\sigma_t^2 = \omega + \alpha y_{t-1}^2 + \beta \sigma_{t-1}^2 \tag{3.16}$$

where $\omega > 0$, $\alpha, \beta \geq 0$ and $\alpha + \beta < 1$. Here the term α controls the impact of past values of the time series on the variance and β controls the impact of past values of the conditional variance on its present value.

It is interesting to note that if we set $\omega = 0$, $\alpha = 1-\lambda$ and $\beta = \lambda$, the GARCH(1,1) model expressed in Eq. 3.16 can be written:

$$\sigma_t^2 \quad = \quad (1-\lambda)y_{t-1}^2 + \lambda\sigma_{t-1}^2 \qquad (3.17)$$

which is equivalent to the EWMA model for the variance described in Eq. 3.12. Hence, though quite different in appearance, the GARCH(1,1) model provides a generalization of the EWMA model.

While GARCH(1,1) processes are the most widely used in practice, there exist a more general class of GARCH(p,q) models. Here the p most recent observations of y_t^2 and the q most recent estimates of σ_t^2 are used in the updated estimate of σ_t^2. The model takes the form:

$$\sigma_t^2 \quad = \quad \omega + \sum_{i=1}^{p} \alpha_i y_{t-i}^2 + \sum_{j=1}^{q} \beta_j \sigma_{t-j}^2. \qquad (3.18)$$

with $\alpha_1, \ldots \alpha_p, \beta_1 \ldots \beta_q \geq 0$, $w > 0$ and $\sum_{i=1}^{max(p,q)} (\alpha_i + \beta_i) < 1$. While we include the general formulation for completeness, we note that models with large values of p and q may not be appropriate for fMRI data sampled at 2s time intervals, though with decreasing TR values they may become increasingly important. The parameters of the GARCH model can be estimated using maximum likelihood, and most statistical software packages have built-in functions for fitting GARCH models.

While many multivariate GARCH models exist that can be used to estimate dynamic correlations, it has been shown that the DCC model out-performs the rest (Engle, 2002a). To illustrate the DCC approach, again assume $\mathbf{y}_t = \epsilon_t$ is a bivariate mean zero time series with conditional covariance matrix Σ_t. The first order form of DCC can be expressed as follows:

$$\sigma^2_{i,t} = \omega_i + \alpha_i y^2_{i,t-1} + \beta_i \sigma^2_{i,t-1} \text{ for } i = 1, 2 \tag{3.19}$$

$$\mathbf{D}_t = diag\{\sigma_{1,t}, \sigma_{2,t}\} \tag{3.20}$$

$$\epsilon_t = \mathbf{D}_t^{-1} \mathbf{e}_t \tag{3.21}$$

$$\mathbf{Q}_t = (1 - \theta_1 - \theta_2)\bar{\mathbf{Q}} + \theta_1 \epsilon_{t-1} \epsilon'_{t-1} + \theta_2 \mathbf{Q}_{t-1} \tag{3.22}$$

$$\mathbf{R}_t = diag\{\mathbf{Q}_t\}^{-1/2} \mathbf{Q}_t diag\{\mathbf{Q}_t\}^{-1/2} \tag{3.23}$$

$$\Sigma_t = \mathbf{D}_t \mathbf{R}_t \mathbf{D}_t \tag{3.24}$$

The DCC algorithm basically consists of two steps. In the first step (Eqs. 3.19-3.21), univariate GARCH(1,1) models are fit (Eq. 3.19) to each of the two univariate time series that make up $\mathbf{y}_t$, and used to compute standardized residuals (Eq. 3.21). In the second step (Eqs. 3.22-3.24), an EWMA-type method is applied to the standardized residuals to compute a non-normalized version of the time-varying correlation matrix R_t (Eq. 3.22). Here $\bar{\mathbf{Q}}$ represents the unconditional covariance matrix of ϵ_t and (θ_1, θ_2) are non-negative scalars satisfying $0 < \theta_1 + \theta_2 < 1$. Eq. 3.23 is simply a rescaling step to ensure a proper correlation matrix is created, while Eq. 3.24 computes the time-varying covariance matrix. The model parameters $(\omega_1, \alpha_1, \beta_1, \omega_2, \alpha_2, \beta_2, \theta_1, \theta_2)$ can be estimated using a two-stage approach. In the first stage, time-varying variances are estimated for each time series. In the second stage, the standardized residuals are used to estimate the dynamic correlations. This two-stage approach has been shown (Engle and Sheppard, 2001; Engle, 2002a) to provide estimates that are consistent and asymptotically normal with a variance that can be computed using the generalized method of moments approach.

Fig. 3.3 illustrates the estimated conditional correlation fit to null data (i.e. two uncorrelated time courses) using DCC. Clearly the method is able to closely follow the true value and appears to be a good candidate for use with fMRI data.

Due to the parametric nature of the model, confidence bands can be created for the dynamic correlation term by using the first two moments of the DCC parameters to simulate their empirical distribution under the assumption of joint normality. Monte-Carlo methods can then be used to repeatedly draw from the estimated distribution of the coefficients, refit the model and create confidence intervals for the dynamic correlations. Details of the estimation and inference procedure are outlined in the Appendix.

3.2.4 Simulations

To evaluate the performance of the different methods for estimating time-varying correlations, we performed a series of simulation studies. For the first three, we generated random data $y_t = (y_{1,t},\, y_{2,t})^T$ for each time point $t = 1, \dots, T$ using a mean-zero bivariate normal distribution. In each case the covariance matrix of the distribution was set to

$$\Sigma_t = \begin{pmatrix} 2 & p(t) \\ p(t) & 3 \end{pmatrix}$$

where the covariance term, $p(t)$, was allowed to vary across time for $t = 1, \dots, T$. This allowed us to control the dynamic relationship between the two time courses $y_{1,t}$ and $y_{2,t}$ throughout the time series. For each simulation, the value of $p(t)$ and T were set as follows:

1. $p(t) = 0$ for all values of $t = 1, \dots T$. This represents the case where the time series are uncorrelated across the entire time course, i.e. null data. The number of time points T varied between 150, 300, 600 and 1000.

2. $p(t) = sin(t/\Delta)$ for $t = 1, \dots T$, $\Delta = 1024/(2^k)$ and $k = 1, \dots, 4$. This represents a slowly varying periodic change in correlation. In this simulation we fix $T = 600$.

3. $p(t)$ is equal to a Gaussian kernel with mean 250 and standard deviation $15 * k$ for $t = 1, \dots, T$ and $k = 1, \dots, 4$. Hence, $p(t)$ is non-zero in an interval that lies

approximately within ±3 standard deviations of 250. This represents a more transient state change from a state of no correlation to an enhanced state of positive correlation. In this simulation we fix $T = 600$.

For each choice of $p(t)$ and number of time points T, the simulations were repeated 1000 times. In each simulation, we fit the sliding-window technique using varying window lengths of 15, 30, 60 and 120 time points, in order to investigate the method's sensitivity to this parameter choice. We also fit the tapered sliding-window using a window length of 22 time points convolved with a Gaussian kernel with standard deviation of 3 time points, the EWMA approach (using the MLE of λ) and the first-order DCC model to the data. In total we evaluated seven different approaches, counting the four variants of the sliding-window approach.

In Simulation 1 we computed the maximum estimated correlation across time for each method and value of T. The goal was to investigate each method's potential for overstating the correlation found in the null data, with small values preferable to large. In addition, to evaluate the approach for computing confidence bounds for DCC we also computed 99% confidence intervals and evaluated their coverage of 0 across time. This was not performed for the other techniques, as methods for constructing confidence intervals are not readably available for them.

For each repetition of Simulations 1-3 we measured the mean square error (MSE) between the estimated correlation and the true value (i.e., $p(t)/\sqrt{6}$) to quantify the different methods' abilities to effectively track dynamic changes in connectivity. In addition, for each repetition we also computed what we refer to as the "oracle sliding window" (OSW) for comparison with the DCC approach. The OSW is defined as the sliding window whose value of w minimizes the MSE, i.e. that best fits the *true* underlying dynamic correlation. The OSW was computed by searching across all window lengths ranging from 30 to 120 in order to find the optimum. It is important to note that the OSW is an idealized statistic that can only be estimated if the underlying truth is known, and thus it cannot be used on experimental data. However, in the context of a simulation study it allows us to compute how well the sliding window technique can theoretically do if we were to always choose the optimal window length in every given situation, and thus provides us with a "gold standard" for the approach.

Finally, in Simulation 4 we investigate the scalability of the DCC algorithm. Here we set the covariance matrix Σ_t to be an $N \times N$ matrix consisting of three equally sized block diagonal elements. Each of these consisted of an $N/3 \times N/3$ matrix with the value 2 in the diagonals and $p_i(t)$ in the off-diagonal elements, where $i = 1, 2, 3$ represents the three submatrices. The values of $p_i(t)$ were chosen as follows: $p_1(t) = 0$, $p_2(t) = sin(t/512)$ and $p_3(t) = 0.5(1 + sin(t/16))$ for $t = 1, \dots T$, where $T = 600$. For a set of values for N ranging from 6 to 102 (in steps of 6), we generated random data y_t for each time point $t = 1, \dots, T$ using a mean-zero multivariate normal distribution with covariance matrix Σ_t. For each repetition we assessed both the performance of the DCC model based upon the MSE and its computation time.

3.2.5 Experimental Data

We applied the DCC approach to the “Multimodal MRI Reproducibility Resource” (Landman et al., 2011), colloquially known as the Kirby 21 dataset, which is publicly available through the Neuroimaging Informatics Tools and Resources Clearinghouse (www.nitrc.org). The Kirby 21 dataset consists of test-retest MRI scans from 21 healthy adult volunteers with no history of neurological conditions (11 male and 10 female, aged 31.76 ± 9.47 years) that were collected using a variety of modern MRI modalities including structural T1-weighted scans and resting state fMRI scans.

Each resting state scan was 7-min long and was acquired using a single-shot, partially parallel (SENSE) gradient-recalled echo planar sequence with an ascending slice order (TR/TE, 2000/30 ms; FA, 75; SENSE acceleration factor of 2; 3-mm axial slices with a 1-mm slice gap) and an 8-channel head coil. Participants were instructed to relax and fixate on a cross-hair while remaining as still as possible. The two resting state scans were separated by a short break during which the participant exited the scanner.

Image processing was performed using SPM8 and custom MATLAB scripts. Images were registered to the first functional volume and normalized to MNI space using unified segmentation/normalization (SPM8). Functional data were adjusted for slice-time acquisition, as well as participant motion and were transformed to MNI space. Nuisance covariates from white matter and CSF were estimated using CompCor (Behzadi et al., 2007) and regressed from the data along with the absolute and differential motion realignment estimates, and linear trends. Data were then spatially smoothed (6mm kernel). Data from one participant was excluded from further analysis due to a misalignment of the first and second resting state scans.

Following Chang and Glover (Chang and Glover, 2010), a region in the posterior cingulate (3mm-radius sphere, centered at $x = 6$, $y = 58$, and $z = 28$) was selected as the primary ROI for the default-mode network. In addition, the five foci identified in Chang and Glover as showing the greatest variation in dynamic correlation with the posterior cingulate cortex (PCC) were selected. Average time courses were extracted (3mm radius spheres) from these regions which included the right inferior parietal cortex (ROI1, $x = 34$, $y = 58$, and $z = 44$), the right inferior frontal operculum/BA44 (ROI2; $x = 50$, $y = 18$, and $z = 32$), the right inferior temporal cortex (ROI3; $x = 58$, $y = 38$, and $z = 16$), the right inferior orbitofrontal cortex (ROI4; $x = 50$, $y = 26$, and $z = 8$), and the anterior cingulate cortex (ACC)/BA24 (ROI5; $x = 2$, $y = 26$, and $z = 24$).

To evaluate whether the regions exhibited similar temporal activity, the pair-wise correlation coefficients between the raw time series of the 5 ROIs were computed for each scan. As shown in Fig. 3.4, ROIs 1, 2 and 3 appear highly correlated with one another, and distinct from ROIs 4 and 5. This is consistent between both scans. However, contrary to Chang and Glover (Chang and Glover, 2010), the latter two regions do not appear highly correlated in our data set. One possible reason for this discrepancy is that we used the same, relatively small, sized seeds as Chang and

Glover, and it is possible the placement of these seeds do not align perfectly with the regions of interest in our data set. However, since our goal was to get sample time courses to illustrate DCC, and not necessarily replicate the previous study we proceed using these time courses with the caveat that the results obtained using ROIs 1-3 may be more reflective of past work.

The DCC approach was used to estimate the dynamic correlation between the PCC and each of the five ROIs. The range of variability was recorded, as were 99% confidence intervals for the correlation. The latter was used to determine whether the correlations varied significantly over time.

3.3 Results

3.3.1 Simulations

Simulation 1:

For the first simulation, the two time courses were designed to be uncorrelated throughout the entire time course. In this setting we would expect the estimated dynamic correlation to take values close to zero across the entire range of the time course. To illustrate the potential for the different methods to overestimate the correlation, we compute the maximum correlation obtained across time for each method discussed using simulated data of varying length ($T = 150, 300, 600, 1000$). Fig. 3.5 shows results for the sliding-window technique using window lengths of 15, 30, 60 and 120 time points. Clearly for shorter windows this approach consistently gives maximum correlations in the range $0.5-0.6$, and at times even approaches 0.9, based purely on random noise. Hence, using this approach one would expect to get large correlations purely by chance. This result is further confounded by the fact that the correlations will vary smoothly due to the manner in which computations are performed (e.g., see Fig. 3.1), thus providing dynamic time courses that give the illusion of providing important signal, while in reality simply responding to natural variation due to noise. This problem is alleviated by increasing the window length. However, on the flipside this will reduce the method's sensitivity to observe actual correlations differing from 0.

Fig. 3.6 shows similar results obtained using the EWMA and DCC models. For EWMA the maximum correlation lies somewhere between those found using sliding-windows of length 30 and 60. For the DCC approach the maximum correlation observed across time is significantly lower than those reported using the sliding-window technique. It tends to report maximum correlations in the range of $0-0.2$, rarely giving rise to maximum correlations higher than 0.5, and therefore noise will be far less likely to be erroneously interpreted as important signal.

Fig. 3.7 shows the proportion of times the 99% confidence intervals computed using the DCC approach did not cover 0 as a function of time for $T = 600$. Results for other values of T (not shown here) were similar. Clearly, all values lie well below the nominal value of 0.01, and the Monte Carlo procedure appears to have appropriate coverage.

The left panel of Fig. 3.8 shows an example of the true correlation plotted together with the estimated correlations obtained using EWMA, DCC, sliding-windows and tapered sliding-windows for a single repetition of the simulation. The right panel shows boxplots of the mean square error (MSE) for all 1000 repetitions of the simulation when $T = 600$. The MSE is computed by calculating the mean of the squared difference between the estimated dynamic correlation and the true correlation (0). Clearly the DCC model performs very well compared with the other models, as the MSE for the DCC model is an order of magnitude smaller than most of the other methods (i.e. EWMA, tapered sliding windows, sliding windows with lengths $15-60$). In general, as illustrated in the left panel, the estimated correlation tends to remain close to zero across the entire span of the time series, while the other methods tend to oscillate in the range between -0.4 and 0.4. Only the sliding window with length 120 is comparable in performance. This isn't surprising as the truth is a static correlation and including as many points as possible in the computation will improve the results.

Finally, we compared the DCC with the oracle sliding-window, computed using the value of w that minimizes the MSE. Fig. 3.9A shows a boxplot for the percent difference in MSE between DCC and OSW, with negative values indicating lower MSE values for DCC. Clearly, DCC outperforms the OSW in almost all replications, with an average improvement of 0.8%. Fig. 3.9D shows a boxplot of the optimal values of w in the repeated realizations. The most common value is 120, indicating that one should use as much of the data as possible. Since, there is no actual time-varying correlation between time courses, the difference between OSW and DCC will decrease as w approaches T.

Simulation 2:

The left panels of Fig. 3.10 show examples of the true correlations used in this simulation, which are slowly varying periodic functions of differing frequencies. Also shown in the plot are the estimated dynamic correlations obtained using EWMA, DCC, sliding windows and tapered sliding windows for a single repetition of the simulation. The right panels show boxplots of the MSE for all 1000 repetitions of the simulation. For the more slowly varying correlations, DCC and the sliding-windows techniques with longer windows (60 and 120) perform best. However, as the frequencies of the true correlations increase the performance of the sliding-window techniques worsen and DCC comes to perform best.

This can also be seen in Fig. 3.9B where the OSW outperforms the DCC in a majority of repetitions for the more slowly varying correlations, while the opposite results hold as the frequency increases. Interestingly, for these later frequencies the variation of the optimal window length is quite substantial (see Fig. 3.9E), illustrating the inherent difficulties involved in determining an appropriate window length in a real-world situation. These results show that the OSW and DCC perform roughly equivalent to one another. This is particularly remarkable as the OSW uses the true value to calibrate itself. This indicates that even if one were able to always choose the optimal window length in the sliding-window analysis, its performance would not necessarily improve upon that of DCC which is entirely data driven.

Simulation 3:

The left panels of Fig. 3.11 show examples of the true correlation, in this case meant to represent transient state-changes depicted by Gaussian curves of varying width $(15, 30, 45, 60)$, together with the estimated correlations obtained using EWMA, DCC, sliding-windows and tapered sliding-windows for a single repetition of the simulation. The right panels show boxplots of the MSE computed solely over intervals with non-zero correlation (defined as intervals within three standard deviations of the mean located at time point 250) for all 1000 repetitions of the simulation. The MSE was computed in this manner in order to properly evaluate the methods ability to pick up the transient changes, and not be unduly influenced by their ability to detect static null correlation, which was evaluated in Simulation 1. For three out of the four simulations, the DCC model outperforms the other approaches. However, for short state changes (see top row) it performs slightly worse than the EWMA model. This illustrates that the benefits of the DCC model may be less apparent when attempting to estimate rapid state changes, and alternative approaches may be preferable. As the width of the Gaussian increases, the performance of DCC and the sliding-window techniques with longer window lengths (60 and 120) improve, with DCC performing best.

A similar story can be seen in Fig. 3.9C where the performance of the DCC becomes marginally better than the OSW for wider Gaussian kernels. However, as illustrated in Fig. 3.9F the process of determining the optimal window length needed to atain the performance of the OSW promises to be difficult as its value is shown to vary substantially across replications of simulations based upon the same underlying true correlation.

Simulation 4:

The top panels of Fig. 3.12 show the mean and variance of the MSE for each component of the matrix Σ_t for the case when there were $N = 90$ time courses included in the analysis. The results, which are representative of other values of N, are consistent with those seen in the bivariate case. The bottom panel of Fig. 3.12 shows the computation time for DCC as a function of the number of nodes include in the analysis. The results indicate a near linear increase in complexity. On a MacBook Pro (2.4 GHz Intel Core i7) the bivariate case had an average computation time of 1.7s, while for the case with 100 nodes it increased to 63.3s.

3.3.2 Experimental Data

Here we illustrate the application of DCC to the Kirby 21 data set. Fig. 3.13 shows the ranges of the estimated dynamic correlations between the PCC and ROIs 1–5. The ranges vary significantly both across regions and subjects. Interestingly, there is also a large amount of variation across scans for the same region and subject.

We further computed 99% confidence intervals for the correlation and determined how often the intervals did not cover the static correlation coefficient between the PCC and the region in question. The idea is that correlations that consistently cover the static value are less likely to truly vary across time. Fig. 3.14 show heat maps of the number of time points for each region, subject and scan. Again, there appears to be a large amount of variability in the values across regions, subjects and scans.

Fig 3.15 displays data from three illustrative subjects. The first subject demonstrates a small range in connectivity and there are only a few points where the static correlation lies outside of the confidence bounds. The second illustrative subject shows a large range, but there are still only a few points outside of the intervals. Finally, in the third example, the subject has a large range and many points outside of the interval. The estimated correlation is extremely noisy and it leads to the suspicion that the large amount of variation may be driven by noise. In addition, somewhat troubling, the behavior of these correlations are not consistent across repeated scans for the same subjects and regions.

3.4 Discussion

In recent years there has been a growing interest in estimating potential dynamic correlations between two brain regions, as it is thought to provide important information about the properties of brain networks. In this paper we study the behavior of several variants of the commonly used sliding-windows technique and introduce two multivariate volatility models often used in the finance literature to the neuroimaging community.

In general, the paper illustrates that sliding-window techniques may not be particularly useful for tracking dynamic correlations. For short window lengths, our simulations clearly show that even completely random noise will give rise to dynamic profiles that appear to show compelling dynamic changes in correlation across time. This problem is less pronounced as the window length increases, however, this comes at the cost of lowered sensitivity to short-term changes in dynamic correlation. Another issue with sliding-window techniques is that their window length is typically chosen in an *ad hoc* manner, when it would be preferable if it were directly estimated from the data. This is presumably done primarily to reduce computational burden, as methods for automatic window size selection are available in the time series literature. For example, (Ombao and Van Bellegem, 2008) suggest a simple algorithm for coherence estimation, that is a specific application of a more general approach developed by (Lepskii, 1990).

Several of the shortcomings of the sliding-window technique are addressed by using multivariate volatility models that are typically used to study the relationship between the volatilities and co-volatilities of several financial markets. We recommend two such models not currently in use in the neuroimaging community, namely the EWMA and DCC models. Both have found extensive use in the finance literature for modeling time-varying variances and correlations and are generally thought to be preferable to sliding-window type approaches (Bauwens et al., 2006). While the sliding window estimator is non-parametric, both the DCC and EWMA models are parametric. These types of approaches tend to be more powerful as long as the model is reasonably accurate. However, with the usual caveat that if not accurate the more flexible non-parametric methods may be preferable.

EWMA has previously been applied to neuroimaging data to model unexpected changes in the mean BOLD signal across time (Lindquist and Wager, 2005; Lindquist et al., 2007; Wager et al., 2009). In this work the focus is instead on estimating the dynamic nature of the variances of the BOLD signal and the correlations between different regions. EWMA shares some similarities with sliding-window techniques in that the correlation is computed using only data contained within a certain interval of time. However, in EWMA the data points are proportional to their distance from the point t, with weights decaying exponentially (hence the name) as points move further away from the time point of interest. Thus, points are removed from the moving windows in a gradual manner more akin to the tapered sliding-windows approach. The weighting is determined by a parameter λ which can be estimated using maximum likelihood methods under the assumption that the time courses follow a bivariate normal distribution. Hence, EWMA effectively allows the window length to be computed in a data-driven manner tailored to the data at hand.

DCC has to the best of our knowledge never been applied to fMRI data. It shares certain similarities with time series models commonly used to describe fMRI noise, such as the autoregressive (AR) and autoregressive moving-average (ARMA) models (Purdon et al., 2001). DCC allows for a two-stage estimator. In the first stage, time-varying variances are estimated for each time series. In the second stage, the standardized residuals are used to estimate the dynamic correlations. This two-stage approach has been shown (Engle and Sheppard, 2001; Engle, 2002a) to provide estimates that are consistent and asymptotically normal with a variance that can be computed using the generalized method of moments approach. This provides us with a framework for performing inference (e.g., constructing confidence intervals) on both the parameters of the model, as well as, the time-varying correlations themselves.

Our simulations show that the performance of DCC is roughly equivalent to that of the "oracle sliding window", which is allowed to tailor itself to the true underlying correlation. This is remarkable as the OSW is an idealized statistic that can't be computed on experimental data and simply acts as a theoretical gold standard for how well the sliding window technique would perform if it always choose the optimal window length. In short, the results indicate that even if we could always choose the optimal window length we would still not improve upon the results of DCC using the sliding window approach.

Though the focus of this paper is on bivariate correlations, one of the primary benefits of DCC is its scalability. In general when studying covariances in the N-variate case, a total of $N(N+1)/2$ variances and covariances need to be estimated, some of which may be time-varying. In addition, the resulting covariance matrix must be positive definite at each point in time. DCC circumvents these problems by using the above mentioned two stage approach. In the first stage each time series is analyzed separately. Hence, moving from the bivariate to N-variate case necessitates computing an additional $N-2$ univariate GARCH models. However, the complexity of each individual GARCH model is unaffected. In the second stage, there are only two parameters to estimate regardless of the dimension. Hence, this stage scales nicely as well. This near linear increase in complexity is illustrated in Fig. 3.12. On the flipside a potential shortcoming of this approach is that the dynamic behavior of the correlation matrix is only governed by two parameters, which places certain limitations on its time evolution.

In a series of simulation studies, the DCC model was shown to outperform the other studied techniques. This included situations where there was no correlation between time courses and when there was slowly varying periodic fluctuations. DCC had some difficulties with very fast short-term state changes. This is illustrated in Simulation 3 and confirmed in other simulations with very rapid state changes not shown here. In these scenarios, DCC had a tendency to miss the change and instead erroneously estimate it as zero correlation. For these types of situations it is instead our recommendation to use a technique such as Dynamic Connectivity Regression (Cribben et al., 2012, 2013) which may be more appropriate.

DCC was used to study the bivariate dynamic connectivity between PCC, a node in the default-mode network, and 5 ROIS that were identified in the literature due to the fact that their dynamic correlations exhibited a high degree of variability across time. Using test-retest data, we observed a fair amount of variability in the behavior of these correlations across subjects, regions and scans. Though variability between regions and subjects in the same scan is understandable due to the task-free nature of the experiment, the variability across scans for the same subject and region is cause for greater concern. It calls into question the reproducibility of the dynamic correlations, and also our ability to effectively link it to behavioral data.

In conclusion, we believe DCC is an attractive option for effective estimation of dynamic changes in correlation in fMRI data. In contrast to commonly used sliding-windows techniques, we have shown that DCC is not easily confused by changes in correlation induced solely by random noise. In addition, all the parameters are effectively estimated through quasi-maximum likelihood methods and require no *ad hoc* parameter settings. Finally, the asymptotic theory of DCC provides a mechanism for statistical inference that is not readably available when using other techniques for estimating dynamic correlations.

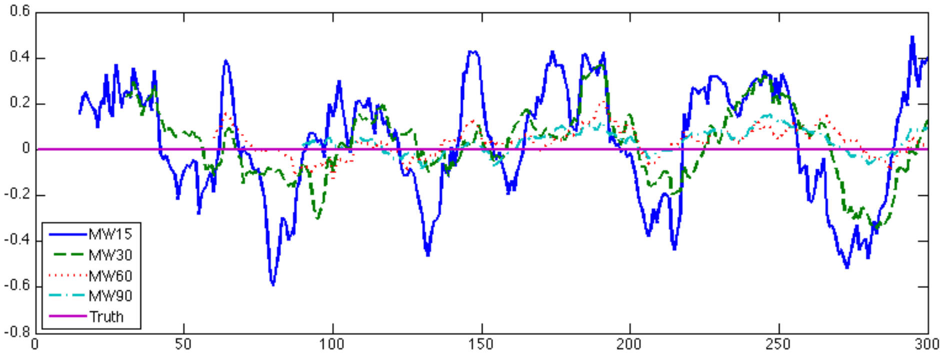

Figure 3.1: Example of the estimated correlation using the sliding windows approach with window lengths $w = 15, 30, 45, 60$. Here the time courses are independent of one another and, hence, there should be no correlation between time courses.

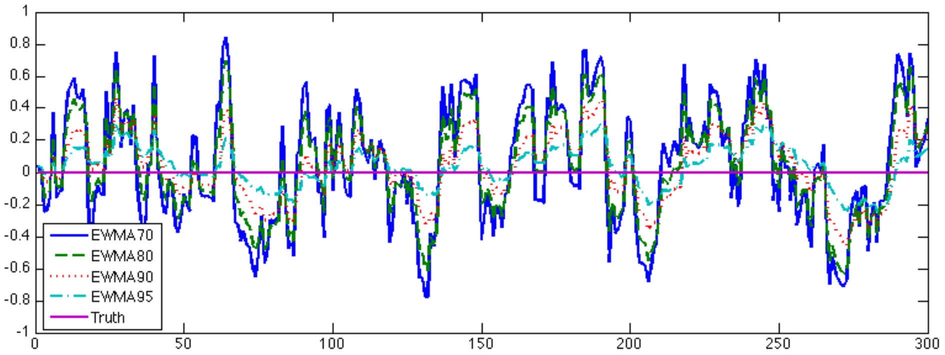

Figure 3.2: Example of the estimated correlation using the EWMA approach with $\lambda = 0.7, 0.8, 0.9, 0.95$. Here the time courses are independent of one another and there should be no correlation between time courses.

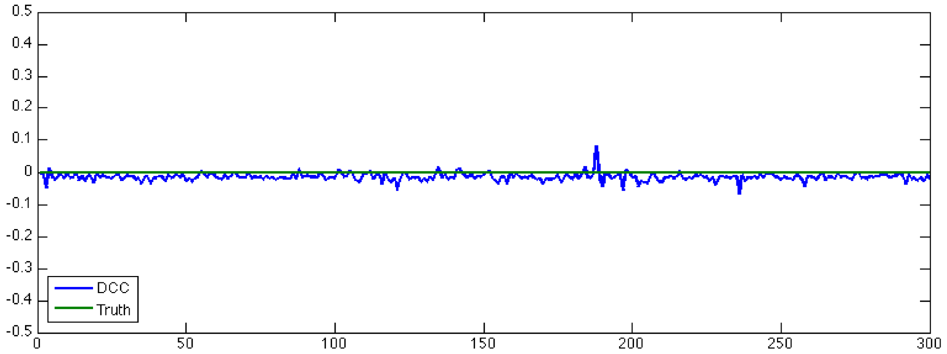

Figure 3.3: Example of the estimated correlation using the DCC model. Here the time courses are independent of one another and there should be no correlation between time courses.

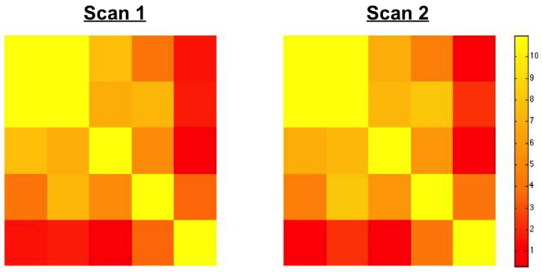

Figure 3.4: The Fisher z-transformed correlation coefficients between the 5 ROIs for both scans. High pair-wise correlation was observed between ROIs 1-3. There also appears to be a high degree of between-scan reliability in the measures.

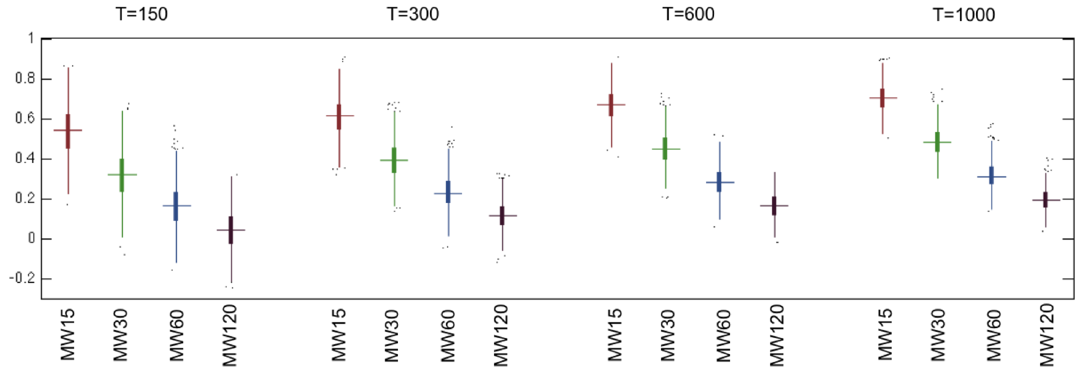

Figure 3.5: Boxplots of the maximum correlation computed using the sliding-window approach with window length $w = 15, 30, 60, 120$ for time courses of length $T = 150, 300, 600, 1000$. In this simulation the time courses are generated to be independent and hence the correlation between time courses should ideally be 0.

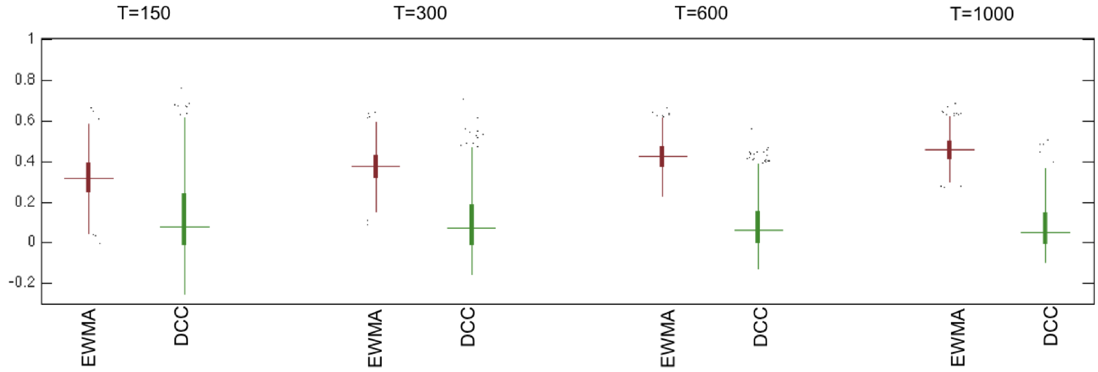

Figure 3.6: Boxplots of the maximum correlation computed using the EWMA and DCC models for time courses of length $T = 150, 300, 600, 1000$. In this simulation the time courses are generated to be independent and the estimated correlation between time courses should ideally be 0.

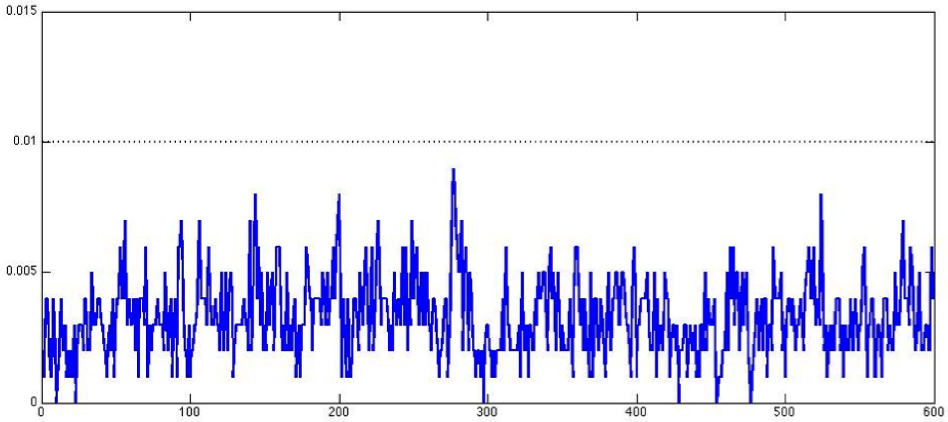

Figure 3.7: The proportion of times the 99% confidence intervals computed using DCC did not cover 0 as a function of time for the case when $T = 600$. Results were similiar for other values of T. Clearly, all values lie below the nominal value of 0.01.

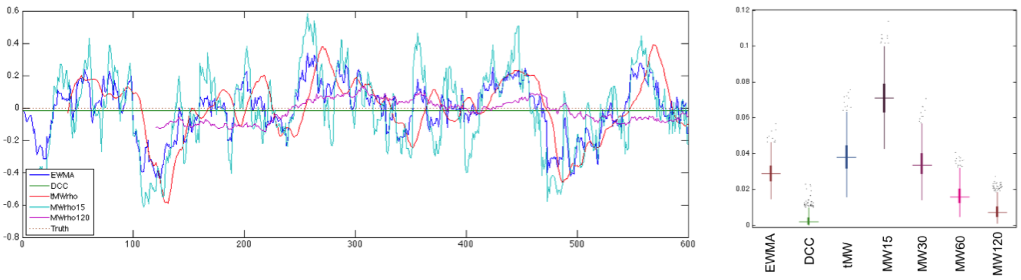

Figure 3.8: Results from Simulation 1 with $T = 600$. The left panel shows the results for a single iteration of the simulation. The truth is no correlation across time and fits for the sliding-window with lengths 15 and 120, the tapered sliding-window, EWMA and DCC are shown overlayed on the plot. The right panel shows a boxplot of the MSE between the estimated dynamic correlations and the truth for each of the seven methods over the 1000 realizations of the simulation.

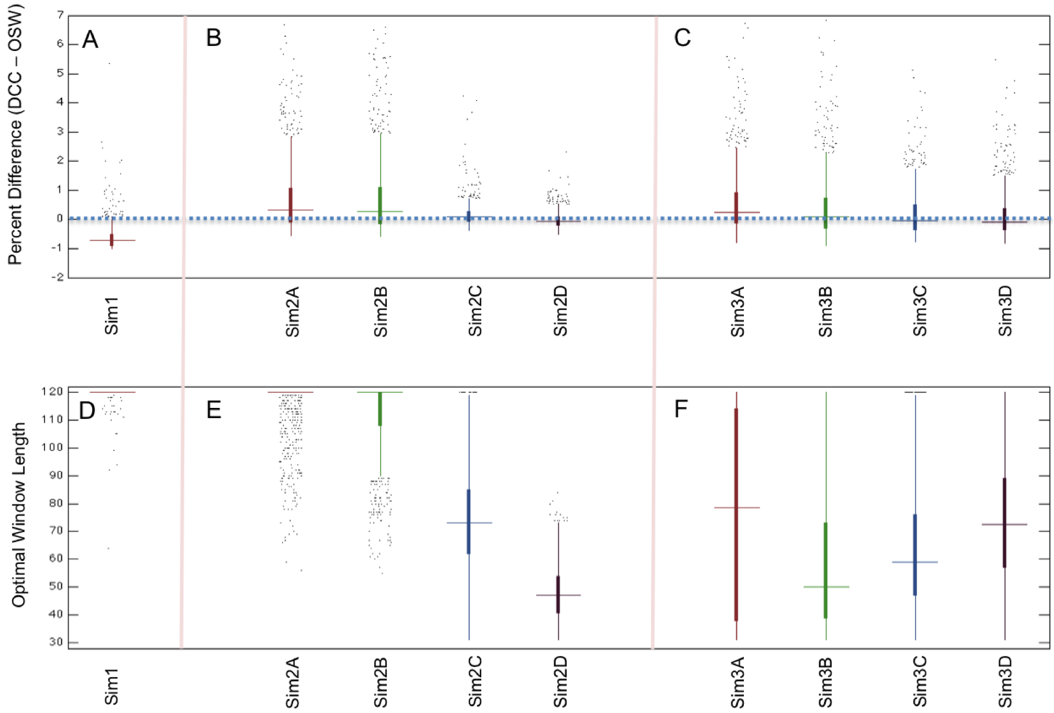

Figure 3.9: A-C. Comparison of the oracle sliding-window and the DCC model for each of the first three simulation studies. The boxplots represent the percent difference in MSE between the two methods based on 1000 realizations of the simulation. As the OSW requires the true correlation to be known it provides a theoretical gold standard for the sliding window approach. D-F. The boxplots show the window lengths that obtained the optimal MSE across the 1000 realizations of each simulation.

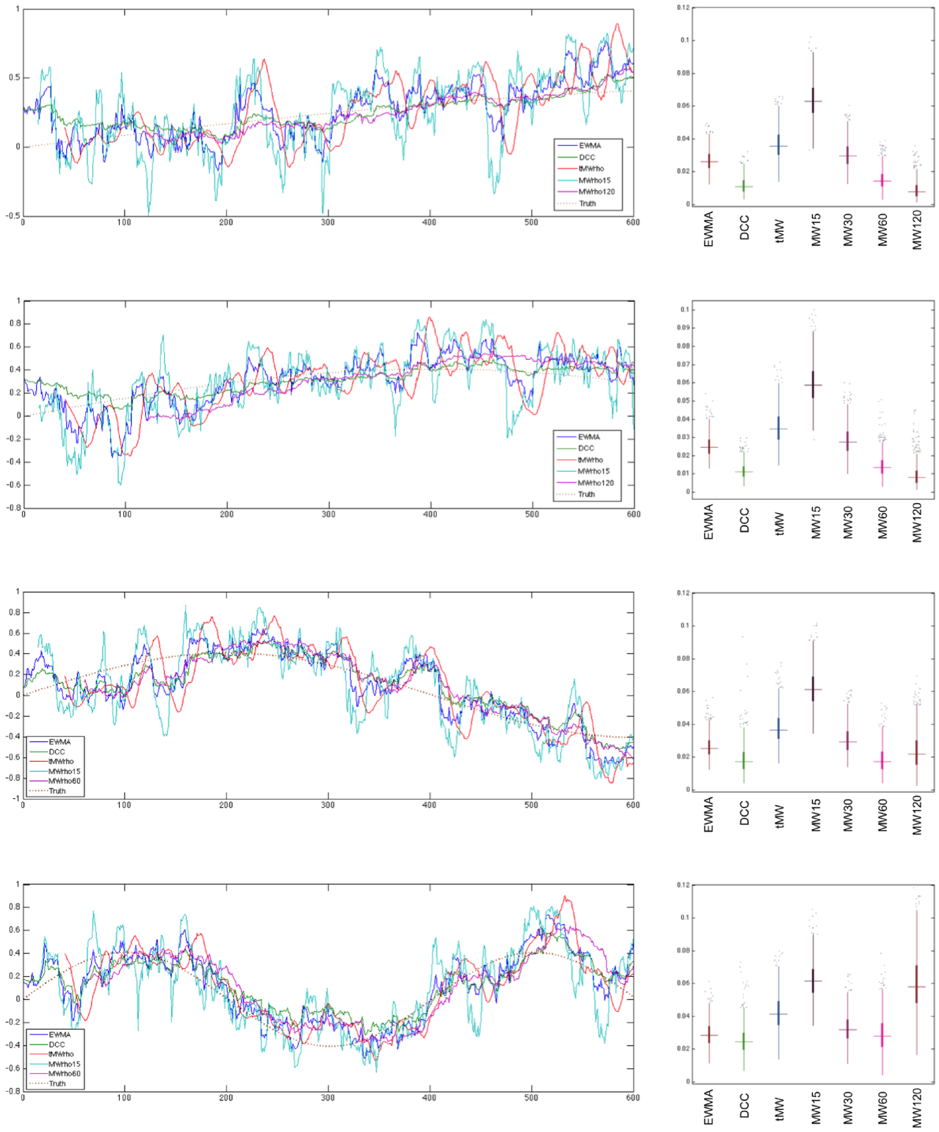

Figure 3.10: Results of Simulation 2. The left panels show the results for a single iteration for each of the four parts of the simulation (one per row). The truth is a slowly varying periodic correlation across time and fits for EWMA, DCC, the tapered sliding-window, and sliding-windows using the worst and best fitting window length are shown in the plot. The right panel shows a boxplot of the MSE between the estimated dynamic correlations and the truth for each of the seven methods over the 1000 realizations of the simulation.

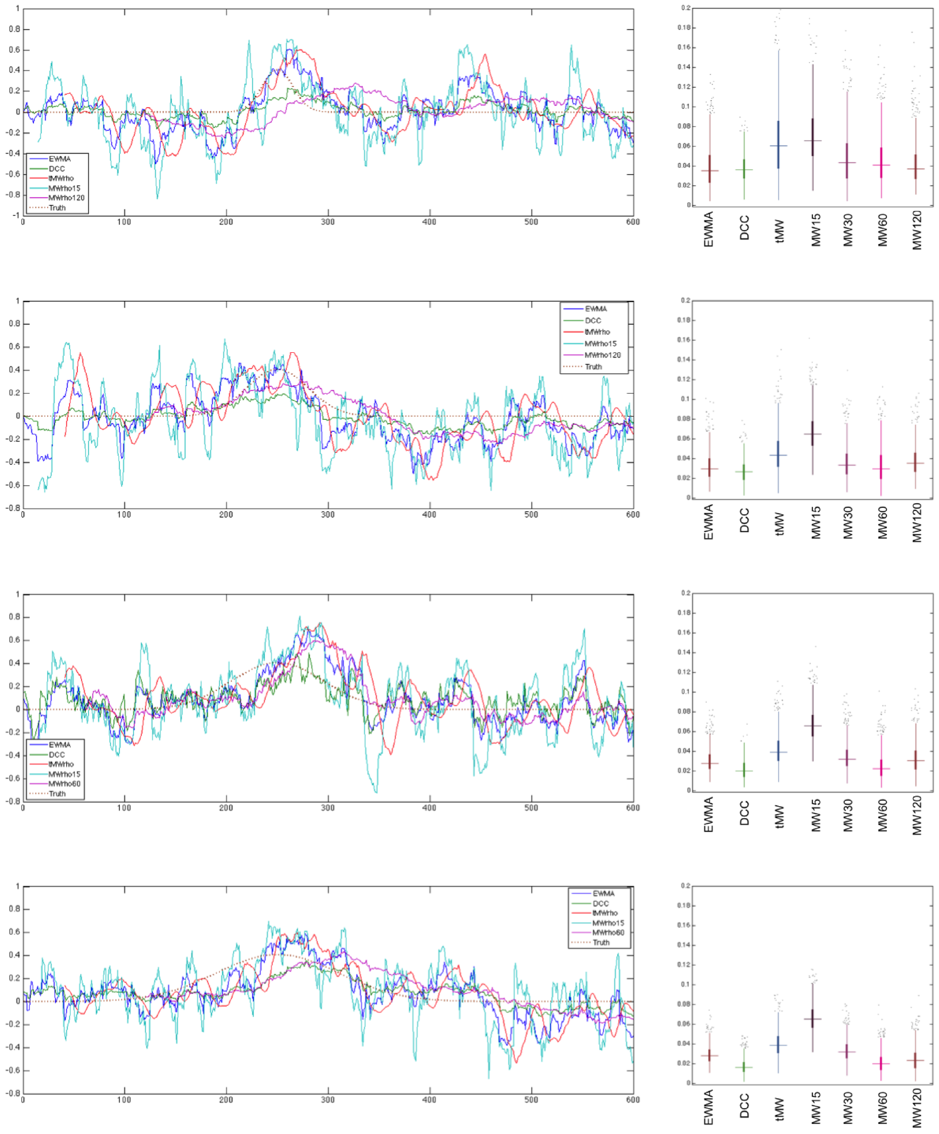

Figure 3.11: Results of Simulation 3. The left panels show the results for a single iteration for each of the four parts of the simulation (one per row). The truth is the correlation is zero except in a range corresponding to a Gaussian curve centered at 250 with standard deviations 15, 30, 45, and 60. The fits for EWMA, DCC, the tapered sliding-window, and sliding-windows using the worst and best fitting window length are shown overlayed on the plot. The right panel shows a boxplot of the MSE between the estimated dynamic correlations and the truth for each of the seven methods over the 1000 realizations of the simulation. The MSE are computed solely over intervals with non-zero correlation, defined as intervals within three standard deviations of the mean located at time point 250.

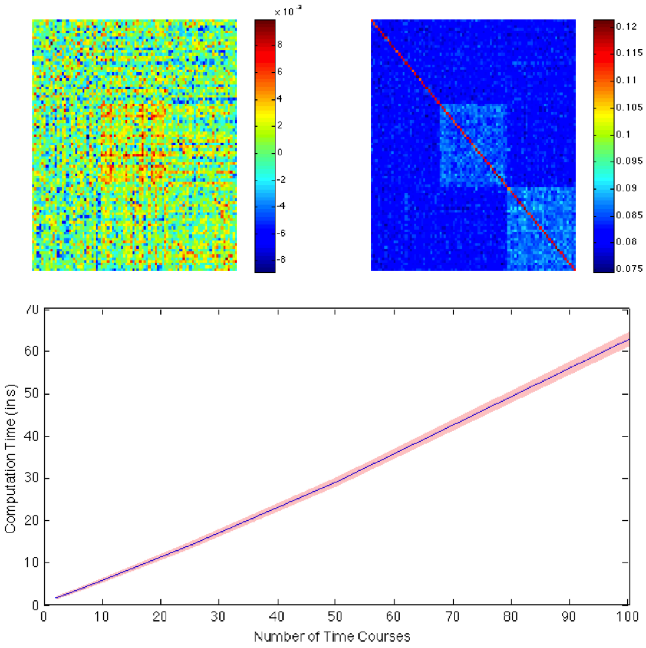

Figure 3.12: Results of Simulation 4. The top panels show the mean (left) and variance (right) of the MSE for each component of the matrix Σ_t when $N = 90$ time courses were included in the analysis. The bottom panel shows the computation time for DCC as a function of N, showing a near linear increase in complexity. The shaded area indicates one standard error around the mean.

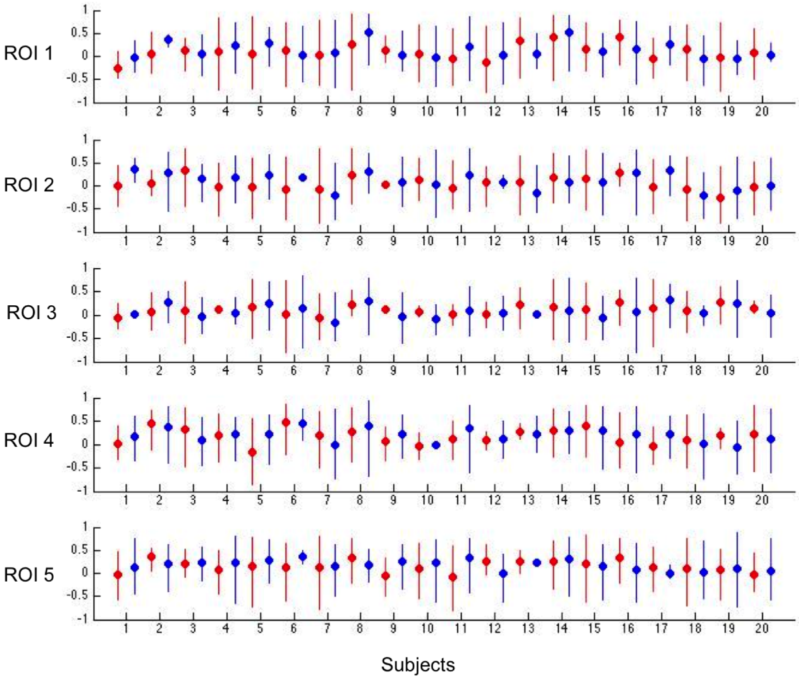

Figure 3.13: The range of dynamic correlation values between the PCC and ROIs 1-5. The vertical lines show the minimum and maximum correlation coefficients across time and the dot indicates the mean. Each subject has two measures, one correponding to scan 1 (red) and the other to scan 2 (blue).

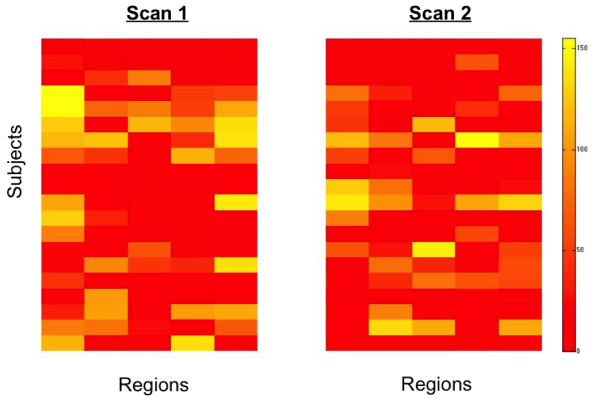

Figure 3.14: The number of time points in which the 99% confidence intervals for the correlation did not cover the static correlation coefficient between the PCC and ROIs 1-5 for each subject, region and scan.

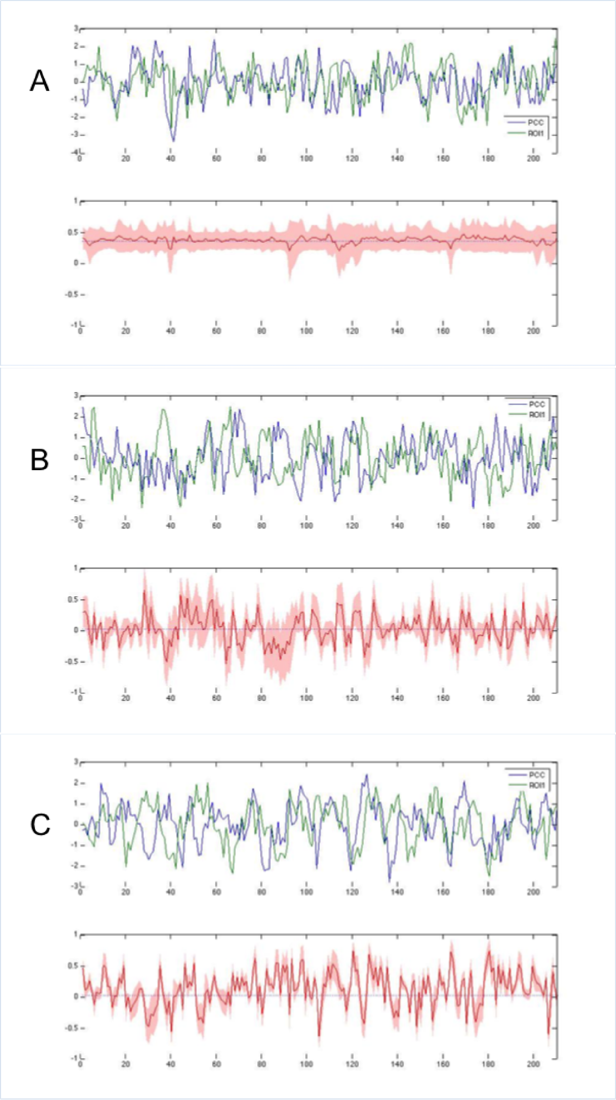

Figure 3.15: The raw time series and estimated dynamic correlations between the PCC and ROI 1 for (A) subject 2 (scan 2); (B) subject 5 (scan 2); and (C) subject 6 (scan 2). The confidence intervals are provided as is the static correlation value (dotted line).

Chapter 4

Moment Estimator for GARCH-DCC Model: an algorithm for dynamic functional connectivity in high dimensional fMRI data

CHAPTER 4. MOMENT ESTIMATOR OF DCC MODEL

Abstract

A significant focus of research on resting-state functional magnetic resonance imaging (rs-fMRI) data has recently begun to shift from the evaluation of static functional connectivity over a scanning session, to the analysis of dynamic functional connectivity. Here the goal is to study changes in functional connectivity on a time-scale ranging from seconds to minutes. In previous work (Lindquist et al., 2014), we proposed the Dynamic Conditional Correlation (DCC) model as an efficient approach for estimating dynamic correlations between rs-fMRI time series from different brain regions. While DCC was shown to be less susceptible to noise-induced temporal variability in correlations than other commonly used approaches in the field (e.g., sliding window techniques), the method becomes biased and computationally expensive as the number of dimensions (i.e. brain regions) increases. In this work we propose a moment-based estimator for DCC (denoted MDCC), along with a fast estimation algorithm, which achieves both more accurate estimation and higher efficiency for high-dimensional time series. We investigate the properties of the newly proposed estimator in various simulation settings, and compare its performance with a recently developed composite likelihood method. The application of MDCC to simulated and real rs-fMRI data, illustrates its efficacy.

4.1 Introduction

During the past decade, the study of human brain organization has been dominated by the use of resting-state functional magnetic resonance imaging (rs-fMRI) data, where blood oxygen level dependent (BOLD) signal is measured in subjects at rest. A number of studies have shown that the brain is in fact not idle during rest, but rather exhibits spontaneous activity that is highly correlated between multiple brain regions (e.g., (Biswal et al., 1997; Fox and Raichle, 2007)). Because of this, the study of functional connectivity (FC), or the undirected association between two or more fMRI time series, has come to the forefront of research efforts in the field.

Recently, focus in the area has begun to shift away from evaluating static functional connectivity over the length of a scanning session, towards analyzing dynamic (or time-varying) functional connectivity ((Hutchison et al., 2013)). Here the goal is to study changes in functional connectivity that take place on a time-scale ranging from seconds to minutes. It is now believed that studying dynamic FC can help provide insight into the fundamental mechanisms of brain function (Hutchison and Morton, 2015; Damaraju et al., 2014; Rashid et al., 2014).

The statistical analysis of dynamic FC in rs-fMRI data is a fundamentally high-dimensional problem with complicated spatial and temporal correlation structure. Here the time series can be obtained either from individual voxels, (Handwerker et al., 2012; Hutchison et al., 2013; Leonardi and Van De Ville, 2015), averaged over pre-specified regions of interest (Chang and Glover, 2010), or estimated using independent component analysis (Allen et al., 2012). Detecting reliable, neuronally relevant changes in FC is difficult due to the low signal-to-noise ratio, physiological artifacts and variation in signal mean and variance over time. It is often unclear whether observed fluctuations in FC should be attributed to neuronal activity or noise (Lindquist et al., 2014). In addition, there remains uncertainty regarding the appropriate analysis strategy to use and how to interpret results (Hutchison et al., 2013; Calhoun et al., 2014).

To date, the most widely used strategy for detecting dynamic functional connectivity has been the so-called sliding window technique (Chang and Glover, 2010), where a correlation matrix is computed over a fixed-length time window of the fMRI time series. The window is thereafter moved step-wise across time, providing a time-varying measure of the correlation between brain regions. Observations within the fixed-length window can be given equal weight, or allowed to gradually enter and exit the window as it is shifted across time (Allen et al., 2012). Potential pitfalls of the family of sliding window methods include the use of arbitrarily fixed-length windows, ignoring values outside of the windows, and an inability to handle abrupt changes in connectivity patterns.

To circumvent some of these shortcomings, multivariate time series models (e.g., Dynamic Conditional Correlations (Engle, 2002a)) that directly model the conditional correlation between fMRI time courses have recently been introduced to the neuroimaging field (Lindquist et al., 2014). These techniques have been shown to be less susceptible to noise-induced temporal variability in correlations compared to the sliding window based approaches. Although the DCC model enjoys the advantages of parsimonious parameterization, simple implementation, and demonstrated success in empirical applications to stock returns (Bauwens et al., 2006; Andersen et al., 2003; Engle, 2002b), it is now well accepted that in the high-dimensional setting the parameters in the DCC model that describe dynamic correlation can be severely biased, which can result in misleading conclusions (Pakel et al., 2014; Engle et al., 2016; Hafner and Reznikova, 2012).

To circumvent this problem a composite likelihood based approach (Pakel et al., 2014) was recently introduced. Using the fact that the DCC model assumes that the time-varying dependency on previous time points is the same for all elements in the correlation matrix, this approach only uses a subset of the data to obtain an approximation of the likelihood needed to estimate the model parameters. When the number of subsets is properly chosen, the estimation should be accurate. Simulation studies indicate that method works well when there are a large number of short series. However, which subsets are chosen and their size is somewhat arbitrary.

In this paper we suggest an alternative approach based of the use of the method of moments instead of the maximum likelihood. Method of moment estimation involves equating sample moments with theoretical moments, and is relatively simple to implement and provides consist estimators under certain weak assumptions. It has efficiently been used in time series analysis to obtain simple estimation equations for the parameters of an autoregressive process (i.e. the Yule-Walker equations).

The paper is organized as follows. In Section 2 we begin by describing the original DCC model, and two maximum composite likelihood methods used for high-dimensional time series. In Section 3 we introduce our new moment estimator for DCC model (denoted MDCC). Thereafter we compare the performance of each method in a series simulation studies, as well as application to real rs-fMRI data. We conclude with a Discussion. The results suggest that the MDCC method can be an efficient and powerful tool for modeling dynamic FC in high-dimensional fMRI data.

4.2 Background

4.2.1 The original DCC approach

The DCC model is a simple class of multivariate Generalized Autoregressive Conditional Heteroskedasticity (GARCH) models that was developed for estimating conditional variances and correlations (Engle, 2002a). The DCC model is easy to implement and has found wide usage in the finance literature (Andersen et al., 2003; Bauwens et al., 2006; Caporin and McAleer, 2012).

To illustrate, let us denote the observed data as $\{\mathbf{r}_t\}_{1\leq t\leq T}$, which is an N-dimensional time series of length T. For simplicity we assume that the time series has been demeaned prior to further analysis. Now, given all the historic information up to time

$t-1$, $\mathcal{F}_{t-1}$, the conditional distribution of $\mathbf{r}_t$ is modeled as:

$$\mathbf{r}_t|\mathcal{F}_{t-1} \sim \mathcal{N}(0, D_t R_t D_t) \tag{4.1}$$

Note that here the conditional variance matrix is decomposed as $D_t R_t D_t$, where

$$D_t = diag(\sqrt{h_{1,t}}, \sqrt{h_{2,t}}, ..., \sqrt{h_{N,t}})$$

is a diagonal matrix with $h_{j,t}$ representing the time-varying volatility, or variance, of the j^{th} time series, and the matrix R_t is the time-varying conditional correlation matrix with diagonal elements equal to one.

In DCC, we begin by first modeling each individual time series using a univariate GARCH processes:

$$h_{j,t} = \theta_{j,1} + \theta_{j,2} r_{j,t-1}^2 + \theta_{j,3} h_{j,t-1}$$

Next, we model R_t as follows:

$$\begin{aligned}
\epsilon_t &= D_t^{-1} r_t \\
Q_t &= S \cdot (1 - \phi_1 - \phi_2) + \phi_1 \cdot \epsilon_{t-1}\epsilon_{t-1}' + \phi_2 \cdot Q_{t-1} \\
R_t &= diag(Q_t)^{-\frac{1}{2}} \cdot Q_t \cdot diag(Q_t)^{-\frac{1}{2}}.
\end{aligned} \tag{4.2}$$

Here S is the sample covariance matrix for ϵ_t, J is an $N \times N$ matrix with all elements equal to one, and $\odot$ is the Hadamard product. Let the parameters in D_t be denoted as θ, and the additional parameters related to R_t be denoted as ϕ. More specifically:

$$\theta = \{(\theta_{j,1}, \theta_{j,2}, \theta_{j,3})\}_{1 \leq j \leq N}, \qquad \phi = (\phi_1, \phi_2)$$

The estimation of the parameters (θ, ϕ) is usually performed in two steps using a quasi-maximum likelihood approach. To illustrate, we begin by expanding the joint log-likelihood of the model $L(\mathbf{r}, \theta, \phi)$ as follows:

$$\begin{aligned} L(\mathbf{r}, \theta, \phi) &= -\frac{1}{2} \sum_{t=1}^{T} (N \log(2\pi) + \log |H_t| + \mathbf{r}_t' H_t^{-1} \mathbf{r}_t) \\ &= -\frac{1}{2} \sum_{t=1}^{T} (N \log(2\pi) + \log |D_t R_t D_t| + \mathbf{r}_t' D_t^{-1} R_t^{-1} D_t^{-1} \mathbf{r}_t) \\ &= -\frac{1}{2} \sum_{t=1}^{T} (N \log(2\pi) + 2 \log |D_t| + \log |R_t| + \epsilon_t' R_t^{-1} \epsilon_t) \\ &= -\frac{1}{2} \sum_{t=1}^{T} (N \log(2\pi) + 2 \log |D_t| + r_t' D_t^{-1} D_t^{-1} r_t - \epsilon_t' \epsilon_t + \log |R_t| + \epsilon_t' R_t^{-1} \epsilon_t) \end{aligned} \tag{4.3}$$

Now the joint log-likelihood can easily be decomposed into two parts, one related to volatility and another related to correlation as follows:

$$L(\mathbf{r}, \theta, \phi) = L_v(\mathbf{r}, \theta) + L_c(\mathbf{r}, \theta, \phi) \tag{4.4}$$

where

$$\begin{aligned}
L_v(\mathbf{r},\theta) &= -\frac{1}{2}\sum_{t=1}^{T}(N\log(2\pi) + 2\log|D_t| + r_t' D_t^{-1} D_t^{-1} r_t) \\
&= -\frac{1}{2}\sum_{j=1}^{N}\Big\{\sum_{t=1}^{T}(\log(2\pi) + \log h_{j,t} + r_{t,j}^2/h_{j,t})\Big\} \qquad (4.5) \\
L_c(\mathbf{r},\theta,\phi) &= -\frac{1}{2}\sum_{t=1}^{T}(-\epsilon_t'\epsilon_t + \log|R_t| + \epsilon_t' R_t^{-1}\epsilon_t).
\end{aligned}$$

In the first step of the estimation procedure we obtain the MLE for θ from the volatility term $L_v(\mathbf{r},\theta)$, i.e.

$$\hat{\theta} = \underset{\theta}{argmax}\, L_v(\mathbf{r},\theta) \qquad (4.6)$$

In the second step, we plug $\hat{\theta}$ into $L_c(\mathbf{r},\theta,\phi)$ and obtain the MLE for ϕ, by computing:

$$\hat{\phi} = \underset{\phi\in\Phi}{argmax}\, L_c(\mathbf{r},\hat{\theta},\phi) \qquad (4.7)$$

where $\Phi = \{x, y \in \mathbb{R} : x \geq 0, y \geq 0, x + y < 1\}$ is the 2-dimensional solution space for ϕ, since for stationary reasons the elements of ϕ must satisfy

$$\phi_1, \phi_2 \geq 0, \qquad \phi_1 + \phi_2 < 1.$$

These constraints guarantee that the covariance matrix H_t and correlation matrix R_t are always positive semi-definite.

DCC has found wide usage, and is generally considered the best option for estimating multivariate volatility models. However, when dealing with high-dimensional time series, the downward bias of the estimator for ϕ is a well-known problem in the econometrics literature (Pakel et al., 2014; Engle et al., 2016; Hafner and Reznikova, 2012). In addition, it is computationally expensive to evaluate the term $L_c(\mathbf{r}, \theta, \phi)$ as it involves computing both the determinant and inverse of the high-dimensional matrix R_t. The focus of this paper is on improving these issues.

The fitting of a univariate GARCH process to each univariate time series is relatively cheap ($O(N)$ time). Therefore, in the continuation we focus on improvements to the estimation of ϕ in the high-dimensional time setting. To do so, we begin by emphasizing an important upgrade of the DCC model, namely the consistent DCC (cDCC) model proposed by Aielli(Aielli, 2013). In this model one replaces Eq. 4.2 by

$$\begin{aligned} \epsilon_t &= D_t^{-1} r_t \\ \epsilon_t^* &= diag(Q_t)^{\frac{1}{2}} \cdot \epsilon_t \\ Q_t &= S \cdot (1 - \phi_1 - \phi_2) + \phi_1 \cdot \epsilon_{t-1}^* \epsilon_{t-1}^{*'} + \phi_2 \cdot Q_{t-1} \\ R_t &= diag(Q_t)^{-\frac{1}{2}} \cdot Q_t \cdot diag(Q_t)^{-\frac{1}{2}} \end{aligned} \tag{4.8}$$

Hence, the corrected standardized residuals ϵ_t^* satifsfies:

$$Var(\epsilon_t^* | \mathcal{F}_{t-1}) = Q_t, \qquad E(\epsilon_t^* \epsilon_t^{*'}) = S \tag{4.9}$$

where the unconditional variance matrix S can be estimated consistently using the sample covariance matrix of ϵ_t^*. In the continuation we will use the cDCC approach instead of the classic DCC approach.

4.2.2 The composite likelihood method

In a working paper by Engle et al. (Engle et al., 2008), a composite likelihood (CL) method was proposed to deal with the computational issues involved with fitting high-dimensional time series. This approach approximates the log-likelihood using a series of bivariate marginal densities, instead of using the full joint density. More specifically, the high-dimensional problem of optimizing $L_c(\mathbf{r}, \hat{\theta}, \phi)$ is addressed by decomposing the original problem into a number of smaller problems defined by selecting a subset of multiple bivariate pairs $\mathcal{P} = \{(j_1, j_2) : 1 \leq j_1, j_2 \leq K, j_1 \neq j_2\}$ and adding their likelihood to obtain an approximation of L_c. Hence, we can write:

$$\tilde{L_c}(\mathbf{r}, \hat{\theta}, \phi | \mathcal{P}) = \sum_{(j_1, j_2) \in \mathcal{P}} L_c(\mathbf{r}_{j_1, j_2}, \hat{\theta}, \phi) \tag{4.10}$$

where $\mathbf{r}_{j,j+1}$ is a two-dimensional time series constructed from a pair of nodes (j_1, j_2) from the set $\mathcal{P}$.

This procedure uses the fact that the DCC model assumes that the time-varying dependency on previous time points is the same for all elements in the correlation matrix. Therefore, we only need to select a subset of data to obtain an approximation of L_c and estimate ϕ. When the number of subsets is properly chosen, the estimation of ϕ should be accurate. However, which subsets to used and how many, is chosen in an *ad hoc* manner. Similar to the notation used in (Engle et al., 2008), the Maximum Composite Likelihood Estimator (denoted MCLE) represents the CL estimator of ϕ based on the use of all unique bivariate pairs, and the Maximum Subset Composite Likelihood Estimator (denoted MSCLE) represents the case where only contiguous pairs are used, i.e.

$$\hat{\phi}^{MCLE} = \underset{\phi \in \Phi}{argmax}\, \tilde{L_c}(\mathbf{r}, \hat{\theta}, \phi | \mathcal{P} = \{(j_1, j_2) : 1 \leq j_1, j_2 \leq K, j_1 \neq j_2\})$$

and

$$\hat{\phi}^{MSCLE} = \underset{\phi \in \Phi}{argmax}\, \tilde{L_c}(\mathbf{r}, \hat{\theta}, \phi | \mathcal{P} = \{(j, j+1) : 1 \leq j \leq K-1\})$$

where K is the cross-sectional time series dimension. Intuitively, since K is large, the MSCLE estimator base on $K-1$ bivariate time series should provide a sufficient approximation to the original MLE of ϕ. According to Pakel et al.(Pakel et al., 2014), the composite likelihood approach gives asymptotically consistent estimator for ϕ, and the computational cost can be largely reduced with particular sampling strategy of the bivariate pairs.

4.3 The Moment Estimation Method

Here we propose a method of moment estimator for ϕ as an alternative to the maximum likelihood estimation procedure outlined in Eq. 4.7. Let us begin by first considering the conditional distribution of the corrected standardized residual in Eq. 4.9:

$$\epsilon_t^* | \mathcal{F}_{t-1} \sim N(0, Q_t)$$

Instead of maximizing the quasi-likelihood of the correlation term $L_c(\mathbf{r}, \hat{\theta}, \phi)$, we instead use the second moment of ϵ_t^* and assume Q_t approximates the expectation of $\epsilon_t^* \epsilon_t^{*'} | \mathcal{F}_{t-1}$.

Define the vectorized second moment of ϵ_t^* as $Y_t = vec(\epsilon_t^* \epsilon_t^{*'})$, which is a vector with $N(N+1)/2$ elements (Note we only need to keep the upper triangle and diagonal terms of the symmetric matrix). Then, we can write

$$\mu_t = E[Y_t | \mathcal{F}_{t-1}] = E[vec(\epsilon_t^* \epsilon_t^{*'}) | \mathcal{F}_{t-1}] = vec(Q_t)$$

We propose to use least square method to estimate ϕ as follows:

$$\hat{\phi}_M = \underset{\phi \in \Phi}{argmin} \sum_{t=1}^{T} \big(Y_t - E[Y_t | \mathcal{F}_{t-1}] \big)^2 \tag{4.11}$$

Since both Y_t and $E[Y_t | \mathcal{F}_{t-1}]$ is recursively calculated from residuals $\epsilon := \{\epsilon_t\}_{1 \le t \le T}$ and Q_t, which all depend on the parameter ϕ, the objective function of the above constrained optimization problem is a very complicate nonlinear function in ϕ and cannot be easily solved. However, we developed a fast computational algorithm to find the solution of $\hat{\phi}_M$, which performs well in many empirical studies.

We first observed that the conditional expectation of Y_t can also be expressed as a linear combination of a "design matrix" X_t and ϕ (see Appendix), i.e.,

$$\mu_t = vec(Q_t) = vec(S) \cdot (1 - \phi_1 - \phi_2) + \phi_1 \cdot Y_{t-1} + \phi_2 \cdot vec(Q_{t-1}) = X_t \cdot \phi \qquad (4.12)$$

Then we use an iterative procedure to solve the optimization problem. With the current estimate $\hat{\phi}^{old}$, we first calculate $Y_t = Y_t(\epsilon; \hat{\phi}^{old})$ and $X_t = X_t(\epsilon; \hat{\phi}^{old})$, then approximate the objective function by:

$$\sum_{t=1}^{T} \big(Y_t(\epsilon; \phi) - \mu_t(\epsilon; \phi)\big)^2 \approx \sum_{t=1}^{T} \big(Y_t(\epsilon; \hat{\phi}^{old}) - X_t(\epsilon; \hat{\phi}^{old}) \cdot \phi\big)^2$$

which is a quadratic function of ϕ. Then we define a mapping $F : \Phi \to \Phi$ as the solution of the approximated constrained optimization problem:

$$F(\hat{\phi}^{old}; \{\epsilon_t\}_{1 \le t \le T}) = \hat{\phi}^{new} = \underset{\phi \in \Phi}{argmin} \sum_{t=1}^{T} \big(Y_t(\epsilon; \hat{\phi}^{old}) - X_t(\epsilon; \hat{\phi}^{old}) \cdot \phi\big)^2 \qquad (4.13)$$

Once we have expressed the approximated objective function as $\phi' A \phi + b' \phi$, the solution $\hat{\phi}^{new}$ can be obtained in O(1) time.

It is clear that our desired estimator $\hat{\phi}_M$ should be the fixed point of the mapping

F within Φ, i.e.

$$\hat{\phi}_M = F(\hat{\phi}_M; \{\epsilon_t\}_{1\leq t\leq T}) \tag{4.14}$$

Below we outline the detailed procedure for obtaining the moment estimator $\hat{\phi}_M$ from the unadjusted residual $\{\epsilon_t\}$ given by computing the first step of the DCC method.

1. Using an initial value of ϕ^0, calculate ϵ_t^* and Q_t according to the cDCC model described in Eq. 4.8.
2. Calculate $Y_t = vec(\epsilon_t^* \epsilon_t^{*'})$, and update the design matrix X_t according to Eq. 4.12.
3. In the n^{th} iteration, solve the mapping $\phi^{new} = F(\phi^n; \{\epsilon_t\}_{1\leq t\leq T})$ according to Eq. 4.13.
4. Generate a random variable r from a $U(0,1)$ distribution, and update $\phi^{n+1} \rightarrow (1-r) * \phi^n + r * \phi^{new}$, and iteratively apply the above procedure until converge.

For most initial values and true parameter values, the sequence ϕ^n generated by the approach described above will quickly converge to the fixed point of the mapping F. Occasionally, when it doesn't converge, we can use other optimization algorithms to solve the equivalent optimization problem:

$$\|\hat{\phi}_M - F(\hat{\phi}_M; \{\epsilon_t\}_{1\leq t\leq T})\|_2^2 = 0.$$

Further details of the derivations are included in the Appendix.

4.4 Simulations

In this section we evaluate the performance of four different DCC estimators in a series of simulation studies. These are the maximum likelihood estimator (MLE), the maximum composite likelihood estimator based on the subset of contiguous pairs (MSCLE), the maximum composite likelihood estimator based on all unique bivariate pairs (MCLE), and the moment estimator we proposed (MDCC).

4.4.1 Simulation Study #1

We conduct a Monte Carlo study to compare the performance of several DCC estimation methods under a variety of time series lengths $T = \{100, 200, 500, 1000, 2000, 3000\}$ and cross-sectional dimensions $K = \{10, 30, 50, 100, 200\}$.

In order to focus on comparing the estimation of the conditional correlation parameters $\phi = (\phi_1, \phi_2)$, we assume that the conditional expectation of the simulated time series is equal to zero, and the conditional volatility for each univariate time series is constant and equal to 1. This corresponds to setting $h_{j,t} \equiv 1$ as in Pakel et al. (Pakel et al., 2014). The residual time series is simulated according to the cDCC model described in Eq. 4.8, with parameters $(\phi_1, \phi_2) = (0.1, 0.8)$. For simplicity, we use the identity matrix as the unconditional correlation matrix. Other potential choices for the unconditional correlation have been explored in previous simulation studies in the literature (Pakel et al., 2014), and they indicate that the different estimators are not sensitive to the choice of the unconditional correlation.

For each pair of dimensions T and K, we repeatedly simulate data $M = 100$ times using the same parameter settings and different random seeds, and compute the four estimators described in the previous sections. Based on the M Monte Carlo replications we compute the bias and the root mean squared error (RMSE) of the estimated parameters $\hat{\phi} = (\hat{\phi}_1, \hat{\phi}_2)$ for each method as follows:

$$bias(\hat{\phi}) = \frac{1}{M}\sum_{m=1}^{M}(\hat{\phi}_m - \phi) \qquad RMSE(\hat{\phi}) = \sqrt{\frac{1}{M}\sum_{m=1}^{M}(\hat{\phi}_m - \phi)^2}$$

For each value of T, we simulate $(T + 200)$ observations and discard the first 200 observations (burn-in) in order to reduce the influence of initial values. For the estimation of the cDCC parameters, the constraints $\phi_1 > 0, \phi_2 > 0$ and $\phi_1 + \phi_2 \leq 1$ are used in the optimization algorithms.

The bias and RMSE are presented in Tables 4.1 and 4.2. Note there is one case ($T = 100$ and $K = 200$) where the MLE results are not reported (shown as NA). This is because the MLE estimation relies on the positive definiteness of the correlation matrix, which is not feasible when $T \leq K$. On the other hand, both the two composite likelihood methods and the new moment estimate performs well in this special case.

From Table 4.1 we observe that the the MLE estimator of ϕ has severe downward bias in many cases, especially when the cross-sectional dimension K is comparable to the time series length T. This finding is consistent with the results observed in previous research (Hafner and Reznikova, 2012; Pakel et al., 2014). Although the negative bias slightly decreases as T increases for fixed K, the bias is non-negligible for typical data sizes used in fMRI studies. For large K $(= 200)$, which is a moderate number of regions of interest in the brain imaging setting, the MLE estimate $\hat{\phi_1}$ has $\sim 20\%$ downward bias even when $T = 3000$, which is longer than most standard fMRI experiments.

When both T and K are very small, all of the estimators are substantially biased since there is not enough information to perform valid inference. For any fixed cross-section size K, the bias in all of the estimates tends to decrease as T increases. In most cases, MSCLE, MCLE and MDCC all have much smaller biases compared to the MLE estimate, and for these three methods the magnitude of the biases largely decreases as T increases, while the trend depends less on the value of K. Although sometimes the bias of MDCC is larger than the two composite likelihood methods, the magnitude is still acceptable and outperforms the original MLE method. The RMSE comparison in Table 4.2 shows that in small to medium sized dimensions, the MDCC estimator has larger variance compared to the composite likelihood methods. The results are expected since likelihood-based methods tend to have better asymptotic efficiency. However, each of MSCLE, MCLE and MDCC substantially outperforms the MLE estimator in terms of RMSE.

The computational efficiency is also of great interest. Figures 4.1 and 4.2 display the run time comparison between the four methods. For each pair of K and T, the average CPU time from the 100 repeated runs of each method is computed. Figure 4.1 shows how the run time increases as a function of T for fixed values of K, and Figure 4.2 shows how the run time increases as a function of K for fixed values of T. All the methods have run time proportional to K^2 and T, besides MSCLE, whose run time linearly increases with K. MSCLE and MDCC have similar computational cost in most cases, although for very large values of K, MSCLE saves time by only using a subset of the data. Both MLE and MCLE have high computational costs in large dimensions, making them largely infeasible in the fMRI setting.

According to theorems found in Pakel et.al (Pakel et al., 2014), both of the composite likelihood methods have $\sqrt{T}-$convergence. This simulation study confirms that there is little difference between MCLE and MSCLE in terms of estimation accuracy. Although MCLE is extremely time consuming, we do not suggest replacing MCLE by MSCLE in all situations. This is because there is a trade-off between computational efficiency and accuracy: MSCLE has substantial information loss because it only uses a subset of K bivariate marginal likelihood functions, while MCLE fully utilizes all K^2 pairs. In this simulation study, where data are generated from the true cDCC model, using only a subset of the data is sufficient for valid inference. In most real data applications, the model is very likely to be mis-specified, and sampling a subset of pairs may not be sufficient to appropriately capture the the behavior of the data. Overall the results of the simulation suggest that all three newly introduced methods outperform the original MLE in the high dimensional setting. They all produce significantly more accurate estimation results, while MSCLE and MDCC have lower computational cost.

Table 4.1: The bias of the estimated parameters (ϕ_1, ϕ_2). The estimators in the comparison are: maximum likelihood (MLE), subset pairs composite likelihood (MSCLE), all pairs composite likelihood (MCLE), and the proposed moment estimator (MDCC). The true model is the cDCC model with parameter $(\phi_1, \phi_2) = (0.1, 0.8)$. Results are based on 100 replications for each dimension setting.

	Bias							
	MLE		MSCLE		MCLE		MDCC	
	ϕ_1	ϕ_2	ϕ_1	ϕ_2	ϕ_1	ϕ_2	ϕ_1	ϕ_2
T	K = 10							
100	-0.0205	-0.1098	-0.0036	-0.1313	-0.0071	-0.0855	-0.0280	-0.1337
200	-0.0115	-0.0322	-0.0015	-0.0461	-0.0015	-0.0362	-0.0041	-0.0762
500	-0.0035	-0.0124	0.0011	-0.0140	0.0003	-0.0153	0.0021	-0.0312
1000	-0.0023	-0.0062	0.0008	-0.0102	0.0006	-0.0094	0.0024	-0.0158
2000	-0.0014	-0.0030	0.0008	-0.0055	0.0008	-0.0050	-0.0025	-0.0040
3000	-0.0014	-0.0026	0.0006	-0.0043	0.0012	-0.0051	0.0024	-0.0150
T	K = 30							
100	-0.0594	-0.1937	-0.0072	-0.0767	-0.0065	-0.0774	-0.0037	-0.1459
200	-0.0287	-0.0455	-0.0018	-0.0314	-0.0012	-0.0355	0.0000	-0.0624
500	-0.0139	-0.0075	-0.0003	-0.0140	-0.0003	-0.0135	0.0021	-0.0298
1000	-0.0086	-0.0016	0.0005	-0.0077	0.0010	-0.0093	0.0021	-0.0181
2000	-0.0064	0.0012	0.0005	-0.0045	0.0007	-0.0051	-0.0016	-0.0075
3000	-0.0054	0.0015	0.0012	-0.0055	0.0013	-0.0057	-0.0021	-0.0068
T	K = 50							
100	-0.0886	-0.4585	-0.0066	-0.0783	-0.0095	-0.0694	-0.0136	-0.0892
200	-0.0431	-0.0733	-0.0013	-0.0359	-0.0020	-0.0320	-0.0049	-0.0424
500	-0.0213	-0.0108	-0.0003	-0.0125	-0.0006	-0.0126	-0.0031	-0.0185
1000	-0.0141	0.0001	0.0000	-0.0077	0.0002	-0.0078	-0.0002	-0.0157
2000	-0.0097	0.0032	0.0016	-0.0069	0.0007	-0.0054	-0.0018	-0.0065
3000	-0.0085	0.0041	0.0010	-0.0049	0.0007	-0.0044	0.0023	-0.0095
T	K = 100							
100	-0.0144	-0.2342	-0.0075	-0.0774	-0.0099	-0.0696	-0.0143	-0.0876
200	-0.0770	-0.2247	-0.0023	-0.0323	-0.0023	-0.0326	-0.0013	-0.0546
500	-0.0356	-0.0259	-0.0007	-0.0124	-0.0005	-0.0126	-0.0028	-0.0172
1000	-0.0230	-0.0030	0.0004	-0.0089	0.0002	-0.0072	0.0018	-0.0152
2000	-0.0162	0.0048	0.0007	-0.0058	0.0008	-0.0060	-0.0030	-0.0032
3000	-0.0133	0.0062	0.0011	-0.0051	0.0008	-0.0048	0.0022	-0.0035
T	K = 200							
100	NA	NA	-0.0095	-0.0740	-0.0090	-0.0711	-0.0165	-0.0579
200	-0.0617	-0.0753	-0.0025	-0.0320	-0.0028	-0.0315	-0.0066	-0.0359
500	-0.0577	-0.0807	0.0000	-0.0132	0.0000	-0.0146	-0.0036	-0.0161
1000	-0.0370	-0.0154	0.0005	-0.0080	0.0001	-0.0073	-0.0023	-0.0096
2000	-0.0248	0.0018	0.0006	-0.0053	0.0007	-0.0051	-0.0031	-0.0024
3000	-0.0201	0.0058	0.0008	-0.0047	0.0006	-0.0043	-0.0017	-0.0050

Table 4.2: RMSE of the estimated parameters (ϕ_1, ϕ_2). The estimators in comparison are: maximum likelihood (MLE), subset pairs composite likelihood (MSCLE), all pairs composite likelihood (MCLE), and the proposed moment estimator (MDCC). The true model is the cDCC model with parameter $(\phi_1, \phi_2) = (0.1, 0.8)$. Results are based on 100 replications for each dimension setting.

	RMSE							
	MLE		MSCLE		MCLE		MDCC	
	ϕ_1	ϕ_2	ϕ_1	ϕ_2	ϕ_1	ϕ_2	ϕ_1	ϕ_2
T	K = 10							
100	0.0281	0.1520	0.0351	0.2321	0.0221	0.1294	0.0602	0.2549
200	0.0161	0.0428	0.0223	0.0842	0.0138	0.0474	0.0607	0.2179
500	0.0081	0.0220	0.0135	0.0361	0.0078	0.0228	0.0382	0.1100
1000	0.0051	0.0121	0.0099	0.0238	0.0056	0.0160	0.0266	0.0730
2000	0.0033	0.0078	0.0066	0.0165	0.0048	0.0114	0.0308	0.0560
3000	0.0031	0.0070	0.0053	0.0125	0.0037	0.0095	0.0301	0.0711
T	K = 30							
100	0.0602	0.2099	0.0213	0.0968	0.0128	0.0855	0.0639	0.2468
200	0.0292	0.0475	0.0130	0.0447	0.0080	0.0407	0.0440	0.1395
500	0.0143	0.0105	0.0071	0.0231	0.0045	0.0172	0.0312	0.0844
1000	0.0089	0.0044	0.0050	0.0155	0.0032	0.0123	0.0517	0.0765
2000	0.0066	0.0033	0.0040	0.0111	0.0024	0.0074	0.0096	0.0242
3000	0.0056	0.0030	0.0031	0.0090	0.0022	0.0071	0.0081	0.0221
T	K = 50							
100	0.0888	0.4758	0.0148	0.0884	0.0128	0.0749	0.0390	0.1729
200	0.0434	0.0749	0.0103	0.0443	0.0066	0.0362	0.0254	0.0854
500	0.0215	0.0124	0.0062	0.0183	0.0039	0.0155	0.0189	0.0516
1000	0.0143	0.0035	0.0043	0.0121	0.0030	0.0105	0.0145	0.0415
2000	0.0098	0.0039	0.0032	0.0094	0.0018	0.0068	0.0074	0.0210
3000	0.0085	0.0045	0.0029	0.0081	0.0018	0.0057	0.0397	0.0509
T	K = 100							
100	0.1069	0.3394	0.0139	0.0877	0.0133	0.0737	0.0320	0.1638
200	0.0771	0.2262	0.0083	0.0392	0.0064	0.0363	0.0334	0.1229
500	0.0357	0.0264	0.0051	0.0170	0.0035	0.0148	0.0170	0.0484
1000	0.0230	0.0040	0.0033	0.0118	0.0023	0.0091	0.0547	0.0858
2000	0.0162	0.0051	0.0029	0.0085	0.0018	0.0069	0.0090	0.0227
3000	0.0134	0.0064	0.0023	0.0068	0.0016	0.0058	0.0070	0.0175
T	K = 200							
100	NA	NA	0.0139	0.0827	0.0111	0.0744	0.0308	0.1109
200	0.0772	0.2204	0.0067	0.0354	0.0055	0.0339	0.0216	0.0775
500	0.0578	0.0809	0.0037	0.0163	0.0032	0.0165	0.0141	0.0421
1000	0.0370	0.0156	0.0029	0.0106	0.0020	0.0089	0.0101	0.0290
2000	0.0248	0.0023	0.0022	0.0073	0.0018	0.0062	0.0078	0.0189
3000	0.0202	0.0059	0.0017	0.0059	0.0014	0.0052	0.0064	0.0170

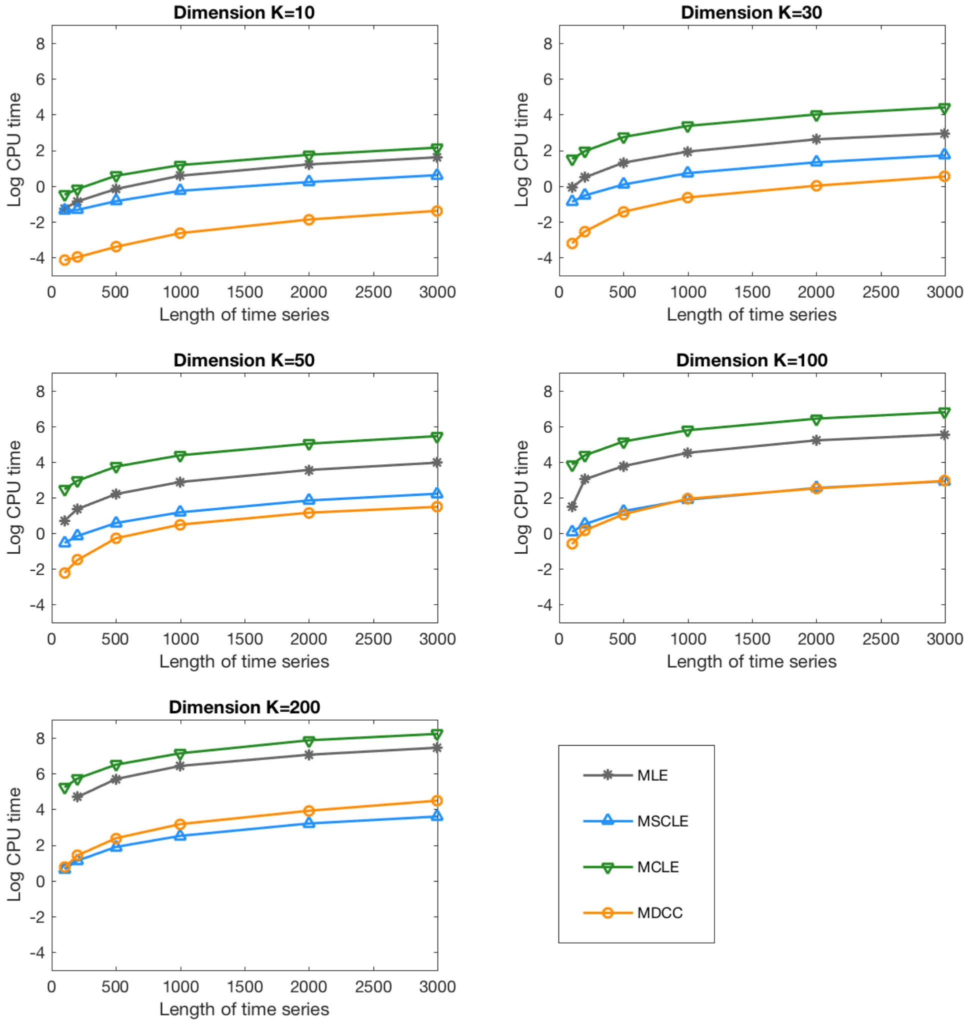

Figure 4.1: Run time comparison for Simulation Study # 1. For each cross-sectional dimension K, the log(CPU time) is plotted against the time series length T. Different line color/marker represents different methods: grey stars are for MLE, blue triangle are for MSCLE, green inverted triangle are for MCLE, orange circle are for MDCC.

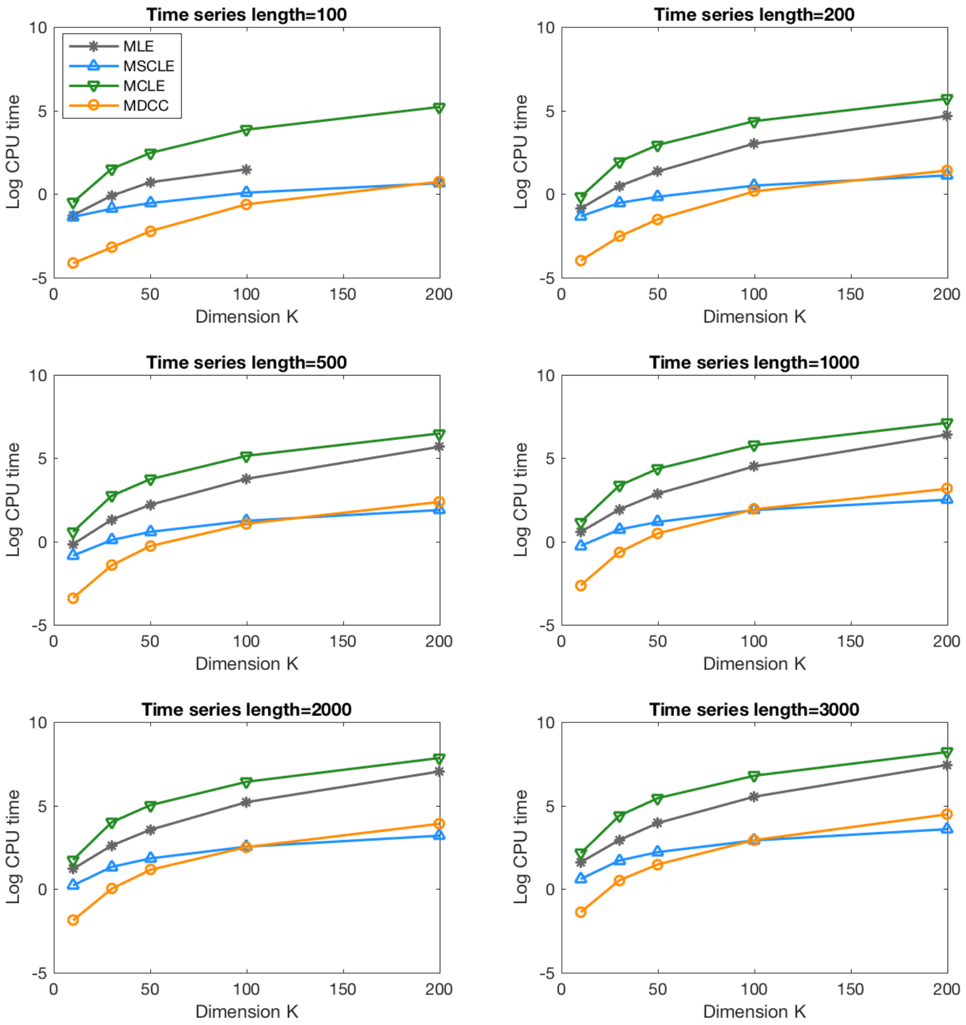

Figure 4.2: Run time comparison for Simulation Study # 1. For each time series length T, the log(CPU time) is plotted against the cross-sectional dimension K. Different line color/marker represents different methods: grey stars are for MLE, blue triangle are for MSCLE, green inverted triangle are for MCLE, orange circle are for MDCC.

4.4.2 Simulation Study #2

Here the performance of the methods are compared in a more realistic setting where the true correlation structure is assumed to be known. In four different cases, we generate data with different types of time-varying correlation matrices, exhibiting constant correlation, sharp changes, and gradual smooth changes. We are interested in comparing how well the estimated correlation matrices $\hat{R}_t$ from the four estimators (MLE, MDCC, MSCLE, MCLE) capture dynamic changes in correlation in high dimensional time series.

While Simulation Study #1 investigated the theoretical properties of the estimators, the comparisons in Simulation Study #2 are more important for practical applications. In reality, the data generating mechanism is unknown, thus the models are usually mis-specified and the theoretical properties of the estimators may not hold in this setting. Moreover, in brain imaging analysis, we are more primarily interested in whether the estimated model can describe the data and capture the essential features in order to study the underlying neurophysiology, rather than simply assess the accuracy of estimate parameters in the ideal setting.

The dimension of the simulated data is fixed throughout Simulation Study #2 with $K = 100$ and $T = 1200$, as this corresponds to the size of the data set used in our application. The residual time series data r_t is generated from the DCC model in Eq.4.1 with different correlation matrices R_t corresponding to each of Cases 1-4. As in Simulation Study #1, we assume away the ARCH effect of volatility by setting D_t

equal to the identity matrix, and focus on assessing the estimation of the correlation matrix:

$$\mathbf{r}_t|\mathcal{F}_{t-1} \sim \mathcal{N}\big(\mathbf{0}, R(t)\big)$$

where the $K \times K$ correlation matrix $R(t)$ is allowed to vary across time. Each element of $R_{ij}(t)$ may differ across pairs (i,j), which allows us to explore the performance of estimators for explaining more complicated dynamic relationship between different regions.

In each case, the true correlation matrix $R(t)$ is specified as follows (the diagonal of $R(t)$ is always 1):

- Case 1. Constant null correlations. $R(t)$ is the identity matrix for $t = 1, \ldots, T$.
- Case 2. Step change. For all non-diagonal pairs (i,j), $R_{ij}(t) = 0$ for $t < T/2$, and $R_{ij}(t) = 0.5$ for $t \geq T/2$.
- Case 3. Slowly varying periodic change. Contains 4 sub-cases with different periodic frequencies. For all non-diagonal pairs (i,j),

$$R_{ij}(t) = 0.9 \times \frac{1 + sin(t/\Delta)}{2}, \qquad \Delta = 1024/2^{\xi}, \quad \xi = \{1, 2, 3, 4\}$$

- Case 4. Slowly varying periodic change with varying magnitude for different pairs.

$$R_{ij}(t) = 0.95^{|j-i|} \times \frac{1 + sin(t/\Delta)}{2}, \qquad \Delta = 1024/2^{\xi}, \ \ \xi = \{1, 2, 3, 4\}$$

For Cases 1 and 2, the model is is simulated $M = 50$ times, and for each of the sub-cases in Cases 3 and 4, the model is simulated $M = 20$ times. For each case and sub-case we compute the RMSE of $\hat{R}(t)$ and the run time of each method for comparison purposes. The RMSE of $R(t)$ is defined as the average RMSE across all pairs and all time points:

$$RMSE\{R(t)\} := \sqrt{\frac{1}{M}\sum_{m=1}^{M}\frac{1}{T}\sum_{t=1}^{T}\frac{1}{K^2}\sum_{1\leq i,j\leq K}\left(\hat{R}^m_{ij}(t) - R_{ij}(t)\right)^2}$$

where $\hat{R}^m_{ij}(t)$ is the estimate of the $(i, j)^{th}$ element of $R(t)$ from the m^{th} run. This metric quantifies the different methods' ability to effectively track the dynamic changes in correlation, with small values preferable to large.

4.4.2.1 Case 1

In Case 1, the time series are uncorrelated across the entire time course, giving rise to null data. The goal is to investigate each method's potential for falsely detecting correlations in the null data. It is preferable that the fluctuations in the estimated correlations be as small as possible.

Figure 4.3 compares the estimated correlations between node pair 1 and 2 using the different methods. In each of the four subplots, the term $\hat{R}^m_{1,2}(t)$ from all the runs $(m = 1, ..., 50)$ are displayed as colored lines, while the true correlation $R_{1,2}(t) = 0$ is shown as a black dashed line. We also illustrate the 5% and 95% point-wise sample quantiles to approximate a 90% confidence interval, displayed as purple dash bands. Each of the methods provide the desired results, as they all estimate small correlations and the estimate tends to be stable, implying that the influence from inaccurate initial values largely decreases over time.

Figure 4.4 compares the RMSE of $R(t)$ across all node pairs and time points for the different methods (left subplot), as well as the computation time (right subplot). The largest RMSE value appears in the MLE method. However, the magnitude is still quite small (< 0.04), which is acceptable. Figure 4.3 illustrates that the performance in accuracy between different methods does not substantially differ, while MCLE requires significantly longer computational time.

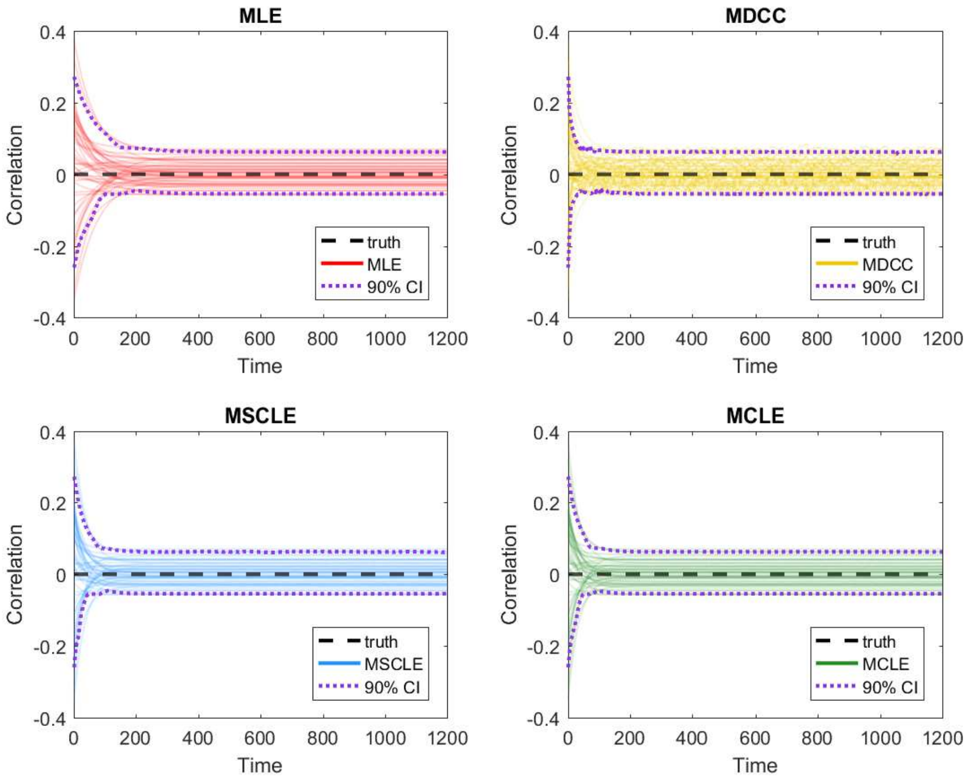

Figure 4.3: Results from Case 1 in Simulation #2. Each panel shows the estimated correlation between node pair 1 and 2 as a function of time for the four different methods. The estimate from all the 50 repeated runs are shown overlaid on each panel, i.e. $\hat{R}^m_{1,2}(t)$ for $m = 1, ..., 50$. The truth is no correlation across time, which is shown as a black dashed line. Point-wise 90% confidence intervals are illustrated using purple dotted lines.

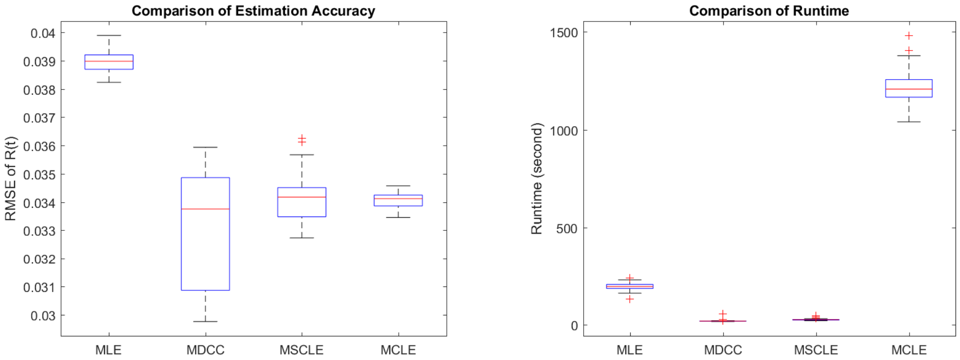

Figure 4.4: Results from Case 1 in Simulation #2. (Left) A boxplot of the RMSE between the estimated dynamic correlations and the truth for each of the four methods from 50 realizations of the simulation. (Right) A boxplot of the computational time in seconds using the four methods.

4.4.2.2 Case 2

In Case 2 the true correlations between all node pairs exhibit a sharp change. During the first half of the time course, there is no correlation, and in the second half, all correlations take a constant positive value (0.5). The sudden increase in correlation happens at $T/2$. Although DCC is better suited for data with gradual changes, we still want to investigate whether the methods could handle the situation with state-related changes, which may exist in task-fMRI time series.

The estimated correlations between node pair 1 and 2 for each method are shown in Figure 4.5. Clearly the original MLE estimator for the DCC model fails to capture the true step change in the high dimensional setting, while the other three methods successfully model the state related change. Based on the 90% confidence intervals, the MDCC estimator performs somewhat worse than the two composite likelihood methods. The estimated magnitude of change is slightly down biased compared to the truth, while the 90% CI for composite likelihood methods cover the true correlation. This can also be seen in the overall RMSE comparison across the four methods shown in Figure 4.6. MDCC, MSCLE, and MCLE, which are specifically designed for high-dimensional data, all have smaller RMSE than the original MLE method. MSCLE and MCLE outperform MDCC in terms of accuracy in that they have consistently lower RMSE across the 50 realizations. However, MDCC and MSCLE are the most computationally efficient methods.

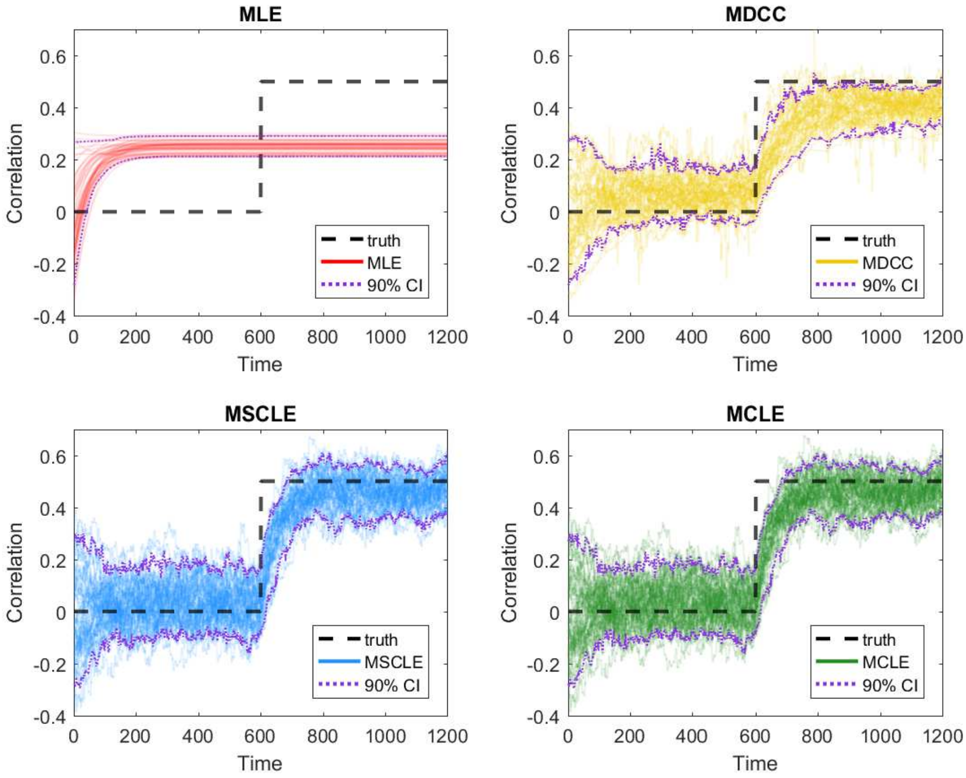

Figure 4.5: Results from Case 2 in Simulation #2. Each panel shows the estimated correlation between node pair 1 and 2 as a function of time for the four different methods. The estimate from all 50 repeated runs are shown overlaid on each panel, i.e. $\hat{R}^m_{1,2}(t)$ for $m = 1, ..., 50$. The truth is a step function with a sudden change in correlation occurring at $T/2$, shown as a black dashed line. Point-wise 90% confidence intervals are illustrated using purple dotted lines.

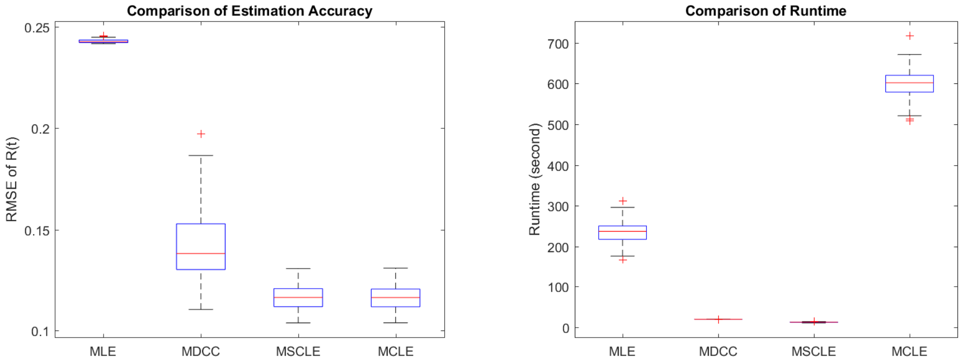

Figure 4.6: Results from Case 2 in Simulation #2. (Left) A boxplot of the RMSE between the estimated dynamic correlations and the truth for each of the four methods from the 50 realizations of the simulation. (Right) A boxplot of the computational time in seconds using the four methods.

4.4.2.3 Case 3

In Case 3 we explore the performance of the different methods for modeling slowly varying dynamic correlations, which are specified as periodic functions of differing frequencies. Figure 4.7 shows the estimated correlation using the four methods between a single pair $R_{1,2}(t)$ for each of the four different frequencies. Similar to the results of Case 2, the MLE method again fails to capture any dynamic changes in the high-dimensional setting. For the case with the lowest frequency (first column), MDCC fails to capture the slight increase in correlation, while the two composite likelihood methods perform well. Here we believe the poor performance of MDCC occurs because, as seen in previous empirical studies (Pang, 1997), in order to describe slow and smooth changes, the "optimal" DCC model should have very small ϕ_1 and large ϕ_2 and thus lies close to the boundary of parameter space. However, for the other three frequencies, MDCC, MSCLE, and MCLE perform similarly and have different pros and cons. MDCC tends to underestimate the variability, but provides smoother estimates of the dynamic correlation, while the composite likelihood based methods capture the changes very well, but exaggerate the variability and provide noisier estimates.

When comparing run time (see Figure 4.8), we find that the MCLE method still has much higher computational costs compared to MDCC and MSCLE, and thus may not be a practical alternative for application to fMRI data.

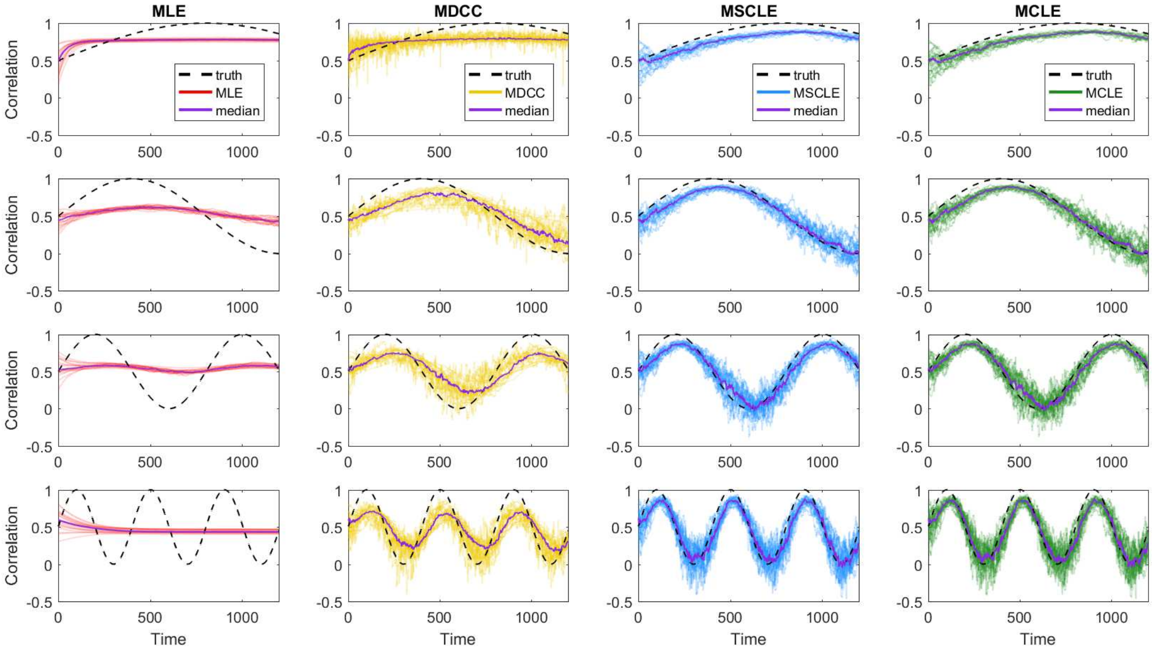

Figure 4.7: Results from Case 3 in Simulation #2. Each panel shows the estimated correlation between node pairs 1 and 2 as a function of time for the four different methods. The truth is a slowly varying periodic correlation across time, shown as black dashed lines. Each row in this figure corresponds to a different frequency of periodic change, and each column corresponds to a different method.The estimates from all 20 repeated runs are overlaid on each panel. The point-wise median of the estimates are shown as purple dotted lines.

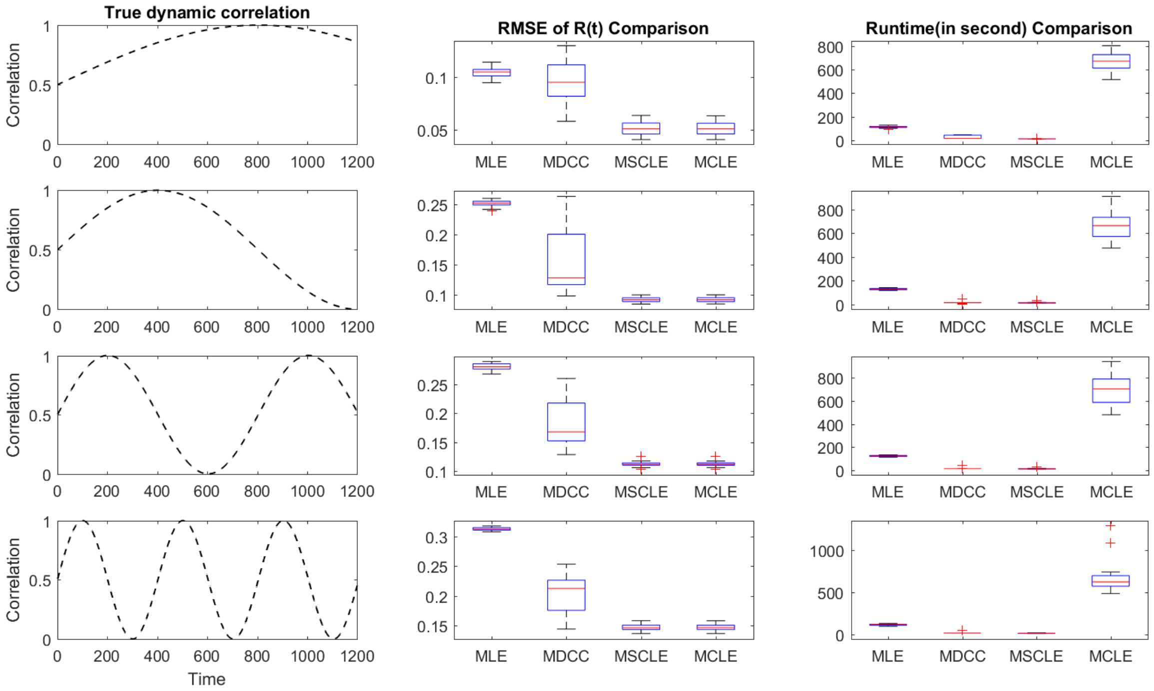

Figure 4.8: Results from Case 3 in Simulation #2. Each row corresponds to a specific frequency of periodic change for the true correlations, shown as black dashed lines in the first column. The middle columns show boxplots of the RMSE between the estimated dynamic correlations and the truth for each of the four methods from 20 realizations of the simulation. The right column shows boxplots of the computational time in seconds using each method.

4.4.2.4 Case 4

Here the setting is similar to Case 3, except for one important difference. The magnitude of the periodic correlation change is allowed to vary between different pairs. Edges between adjacent nodes (i.e., elements close to the diagonal of the correlation matrix) have the largest amount of fluctuation over time, with exponentially decreasing magnitude as the corresponding element in the correlation matrix moves away from the diagonal.

Figures 4.9, 4.10 and 4.11 show results for three pairs of edges $\hat{R}_{1,2}(t)$, $\hat{R}_{1,20}(t)$ and $\hat{R}_{1,50}(t)$ for the same frequencies as in Case 3, using the four different methods. The MLE method again fails to appropriately capture the dynamic changes in correlation. The performance of the two composite likelihood methods differ significantly in this simulation. This is because MSCLE only considers a subset of edges, and here the subset is chosen as pairs of adjacent nodes, which happens to have the largest amount of variability in this simulation. In contrast, MCLE accounts for all bivariate pairs' in the likelihood function. When fluctuations differ between pairs, we expect that both MCLE and MDCC will find the appropriate balance to fit all data, and MSCLE will fit the dynamic correlations of the chosen subset of K pairs very well but perform poorly for the majority of other $O(K^2)$ pairs. The results in Figure 4.9 – 4.12 support our hypothesis. MSCLE performs poorly for most pairs and has the highest overall RMSE of $R(t)$. It gives best estimate for the term $R_{1,2}(t)$ but overestimates the dynamic correlations for other pairs. Although MCLE seems to be the most accurate, its computational costs are overwhelming. In this case, MDCC is the most favorable approach, as it achieves an appropriate balance between accuracy and computational efficiency.

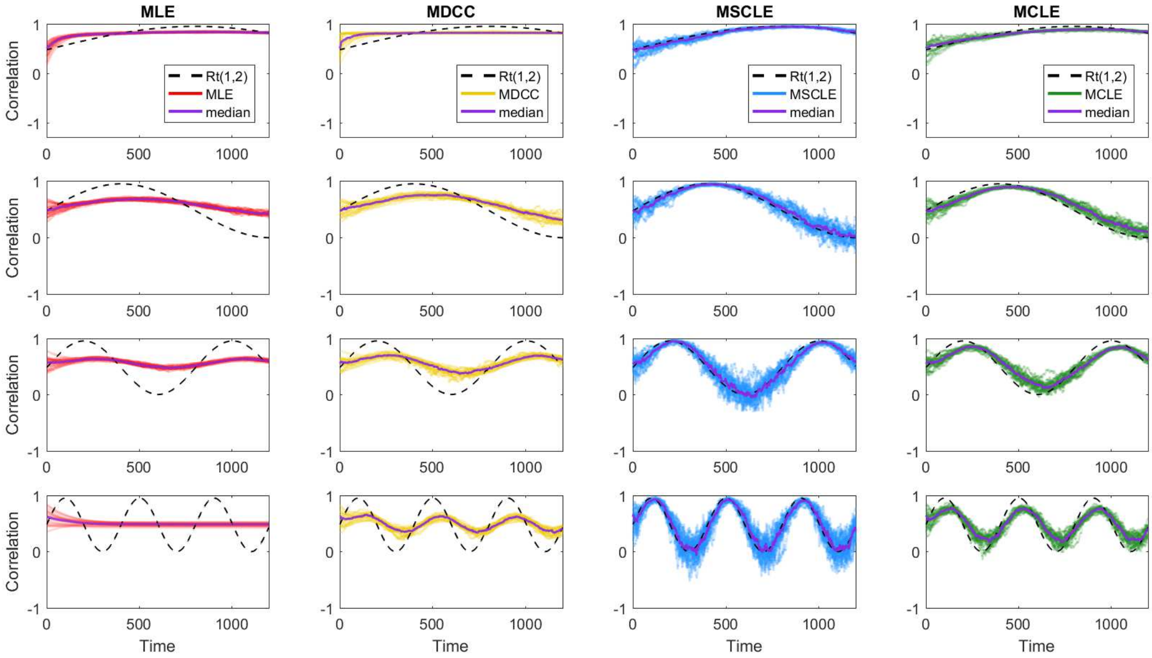

Figure 4.9: Results from Case 4 in Simulation #2. Each panel shows the estimated correlation between node pair 1 and 2 as a function of time using each of the four different methods. The truth is a slowly varying periodic correlation across time with different frequencies, shown as black dashed lines. Each row corresponds to a specific frequency of periodic change of the true correlation, and each column corresponds to a specific method. Estimates from all 20 repeated runs for each frequency are overlaid on each panel.The point-wise median of the estimates are shown as purple dotted lines.

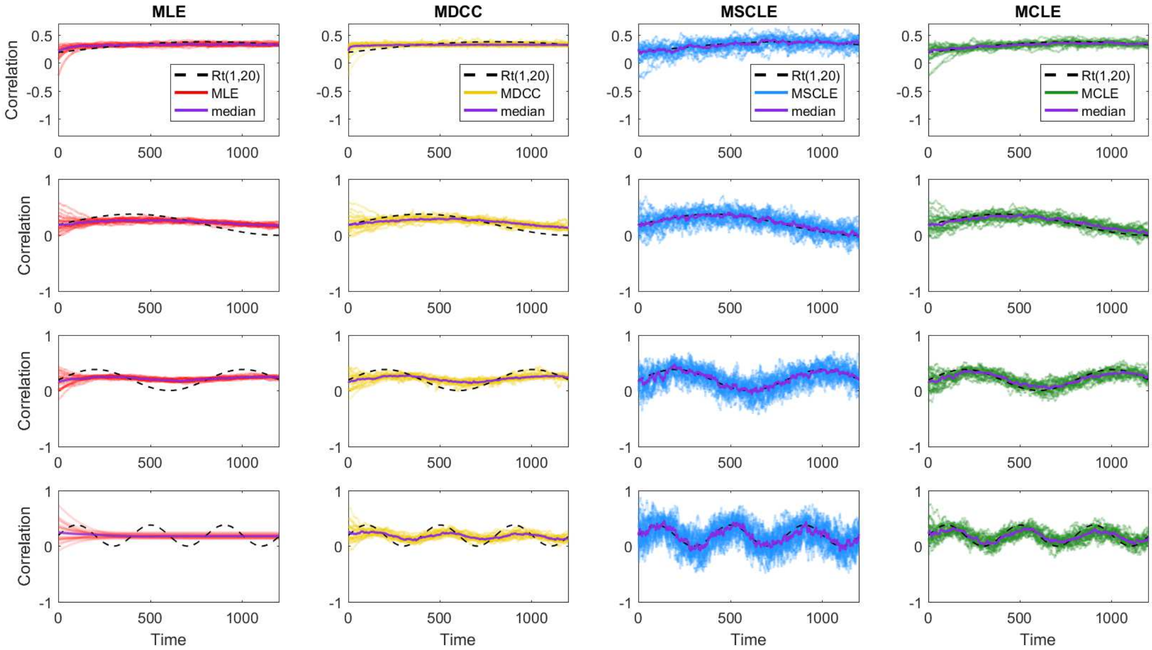

Figure 4.10: Results from Case 4 in Simulation #2. Each panel shows the estimated correlation between node pair 1 and 20 as a function of time using each of the four different methods. The truth is a slowly varying periodic correlation across time with different frequencies, shown as black dashed lines. Each row corresponds to a specific frequency of periodic change of the true correlation, and each column corresponds to a specific method. Estimates from all 20 repeated runs for each frequency are overlaid on each panel.The point-wise median of the estimates are shown as purple dotted lines.

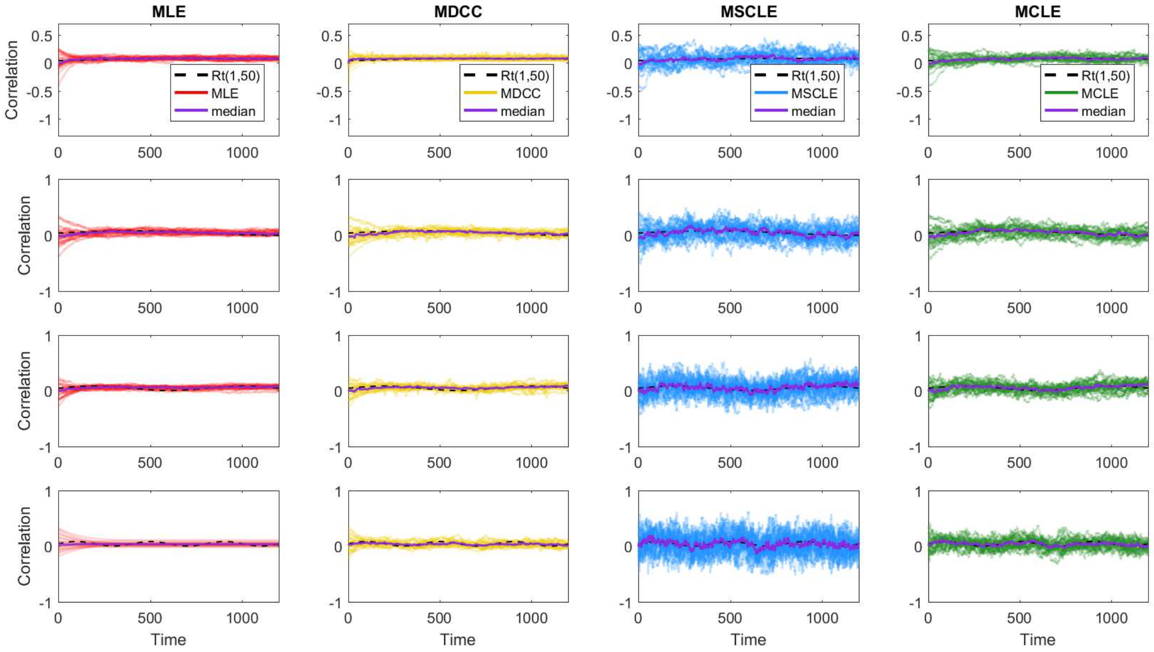

Figure 4.11: Results from Case 4 in Simulation #2. Each panel shows the estimated correlation between node pair 1 and 50 as a function of time using each of the four different methods. The truth is a slowly varying periodic correlation across time with different frequencies, shown as black dashed lines. Each row corresponds to a specific frequency of periodic change of the true correlation, and each column corresponds to a specific method. Estimates from all 20 repeated runs for each frequency are overlaid on each panel.The point-wise median of the estimates are shown as purple dotted lines.

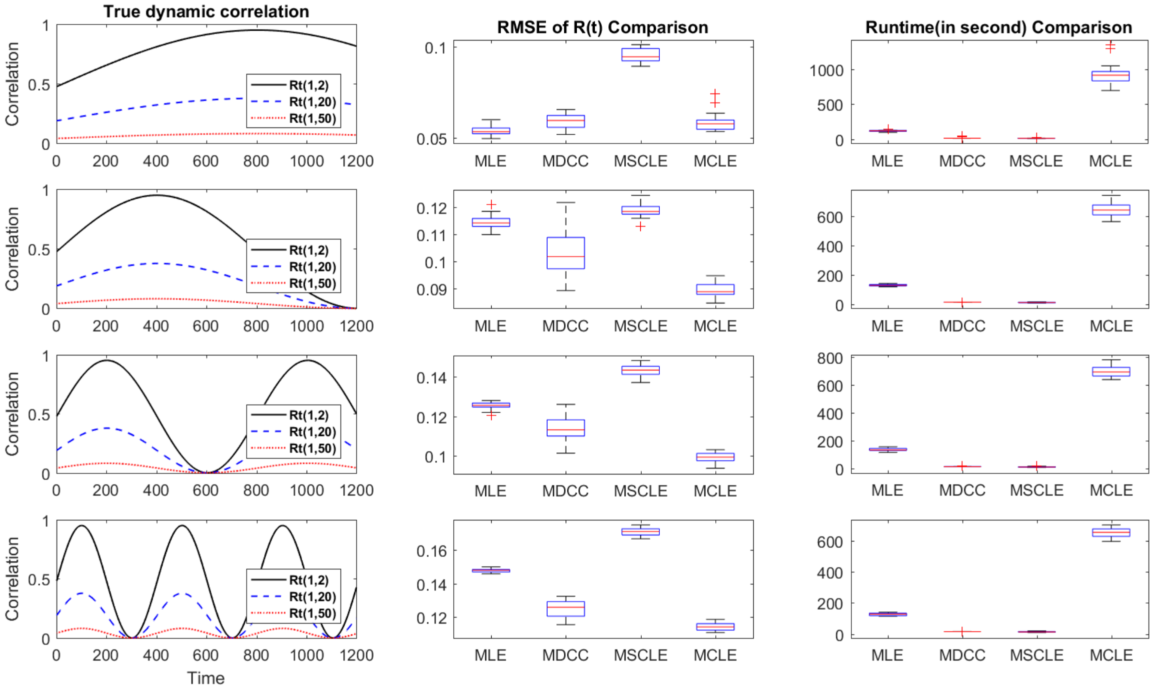

Figure 4.12: Results from Case 4 in Simulation #2. Each row corresponds to a specific frequency of periodic change for the true correlations, shown as black dashed lines in the first column. In contrast to Case 3, the magnitude of the true correlations vary across pairs. The first column shows the three exemplar correlations over time for each frequency.The middle columns show boxplots of the RMSE between the estimated dynamic correlations and the truth for each of the four methods from 20 realizations of the simulation. The right column shows boxplots of the computational time in seconds using each method.

4.5 Application

We used the 2014 Human Connectome Project 500 Parcellation Timeseries Netmats (HCP500-PTN) data set to demonstrate the utility of the different estimators of the DCC model for high-dimensional fMRI time series. The data resource is publicly available at `http://humanconnectome.org`.

The resting-state fMRI data were collected for 461 healthy adults that completed $J = 4$ repeated scans. Each scan lasted for 14 min 24 sec, and consisted of 1200 brain image volumes sampled every 0.72 seconds (i.e., $T = 1200$). Van Essen et al. (Van Essen et al., 2013) provide a detailed explanation of the entire acquisition protocol. The rs-fMRI data from each scan was minimally preprocessed according to the procedure outlined in (Glasser et al., 2013; Smith et al., 2013), and artifacts were removed using the Oxford Center for Functional MRI of the Brain's (FMRIB) ICA-based X-noiseifier (ICA + FIX) procedure (Griffanti et al., 2014; Salimi-Khorshidi et al., 2014). Each scan was temporally demeaned and variance normalized (Beckmann and Smith, 2004). All four scans for the 461 subjects were fed into MELODIC's Incremental Group-Principal Component Analysis (MIGP) algorithm, which estimated the top 4500 weighted spatial eigenvectors. GICA was applied to the output of MIGP using FSL's MELODIC tool (Beckmann and Smith, 2004) using five different dimensionalities (i.e., number of independent components; 25, 50, 100, 200, 300). For each dimensionality, the dual-regression approach was used to map group-level ICA spatial components onto each subject's time series data (Filippini et al., 2009), to obtain a single time series per ICA component.

For illustration purpose, we limit our analyses to the first 20 ($I = 20$) subjects, which includes $20 * 4 = 80$ scans in total. And we choose to use the data set with 100 independent components, i.e. the cross-sectional dimension of the time series in each scan is $K = 100$. We apply the four different estimation methods for the DCC model to compare their performance for modeling dynamic FCs in high-dimensional fMRI time series ($T = 1200$, $K = 100$).

We perform the first step of the DCC estimation by estimating an univariate AR(3)-GARCH(1,1) model for each of the time series, in order to remove the conditional mean and obtain the standardized residual time series for modeling the high-dimensional correlations in the second step. Next, we applied the four alternative estimation methods (MLE, MDCC, MSCLE, MCLE) to the residual time series of each subject i and scan j, and obtained the estimated dynamic correlation matrix $\hat{R}_t^{i,j}(t = 1, ..., T)$ corresponding to each method.

For comparison purposes, we also fit a series of bivariate DCC models to each pair of time series. As the cross-sectional dimension is $K = 100$, this requires us to fit $K * (K - 1)/2 = 4950$ bivariate DCC models to study all the dynamic FC between all pairs. We gather the place the dynamic correlations from each pair into a $K \times K \times T$ matrix to facilitate comparison and visualization. Thus for each time point we have a $K \times K$ matrix summarizing the complete sets of pairwise correlation at time t. However, it should be noted that this is not a correlation matrix in the strict sense, as it is not constrained to satisfy the positive definiteness requirement. Moreover, fitting a large numbers of marginal models is fundamentally different from assuming a joint model for the entire high-dimensional time series.

Using the estimated dynamic correlations from each method, we explore and compare two basic summary statistics for pair-wise dynamic FC, namely the average and standard deviation (SD) of the dynamic correlations within each scan for each subject. The average gives similar information as the standard sample correlation coefficient, which is the average amount of correlation between each pair over time. The standard deviation (SD) of the dynamic correlation can be used to more directly access the FC dynamics. If an edge is involved in frequent state-changes (i.e., exhibits greater FC dynamics), it should exhibit consistently higher fluctuations in correlation across time when compared to edges whose FC remains more static throughout an experimental run.

Figures 4.13 and 4.14 display scatter plots of the average and SD of the dynamic correlation within a scan for all subjects and scans. The results shown in Figure 4.14 suggest= that for most of pairs, the amount of variability in FC is very small, so the connectivity is less likely to truly vary across time. The results using a single DCC model shown in Figure 4.13 further supports this observation. Furthermore, we find more similarities between Figure 4.13 and the results of Case 1 in Simulation #2, where the correlations are constant across time. Recall, when there is no changes in correlation over time, the four estimation methods perform very similarly, although the MLE method exhibits higher fluctuations.

Figures 4.15 and 4.16 display the average and SD of the dynamic correlations estimated by each of the four different methods, where each pair of the two summary statistics are averaged across all subjects/scans. We find that the patterns of FC between regions are very similar using the four different methods, which agrees with our the previous statement.

MDCC is not ideal for low dimensional bivariate data, since it produces higher fluctuations in FC than MLE (Figure 4.13). However, the patterns of FCs in Figure 4.16 obtained using MDCC and MLE are still very similar, which implies that the MDCC method successfully captures edges with higher variability, or connectivity, with lower computational cost.

The computational time for a single scan ($T = 1200$, $K = 100$) using the four different methods are listed in Table 4.3 for comparison. By using MDCC, we reduce the computational cost by approximately $1841.7/23.3 = 79$ times, and obtain a single DCC model for describing the joint-distribution of the high-dimensional data, which enjoys the benefit of parsimonious parametrization as well as providing a more interpretable correlation matrix for each time point. Although MSCLE performs similarly in this case, it should be used with caution, as it basically assumes that the dynamic FCs behaves similarly between all pairs, because it assumes a small subset of pairs is representative of the complete data set.

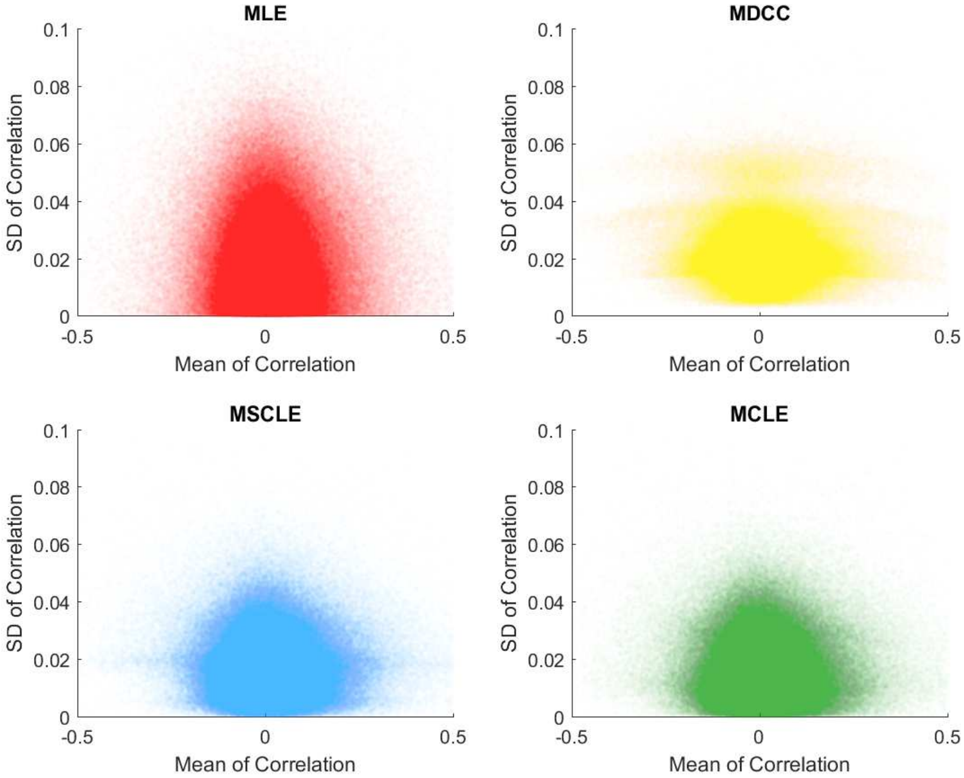

Figure 4.13: Application to HCP data. The standard deviation of the dynamic correlation plotted against the corresponding average, where the dynamic correlations of all edges are estimated together in a single DCC model. The results using the four different methods (MLE, MDCC, MSCLE and MCLE) are displayed in separate panels.

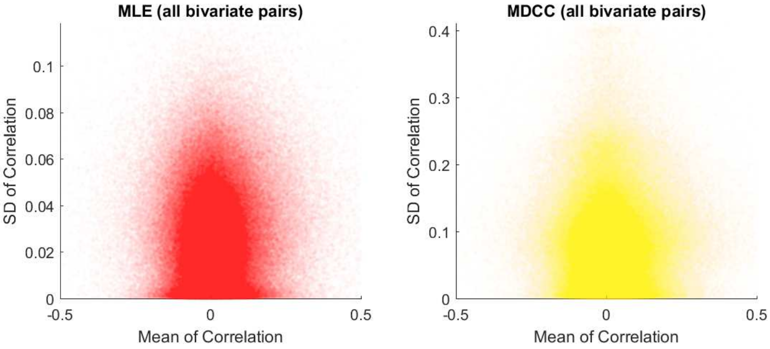

Figure 4.14: Application to HCP data. The standard deviation of the dynamic correlation plotted against the corresponding average, where the dynamic correlation of each edge are estimated separately using bivariate DCC models. The result using MLE and MDCC are displayed in separate panels.

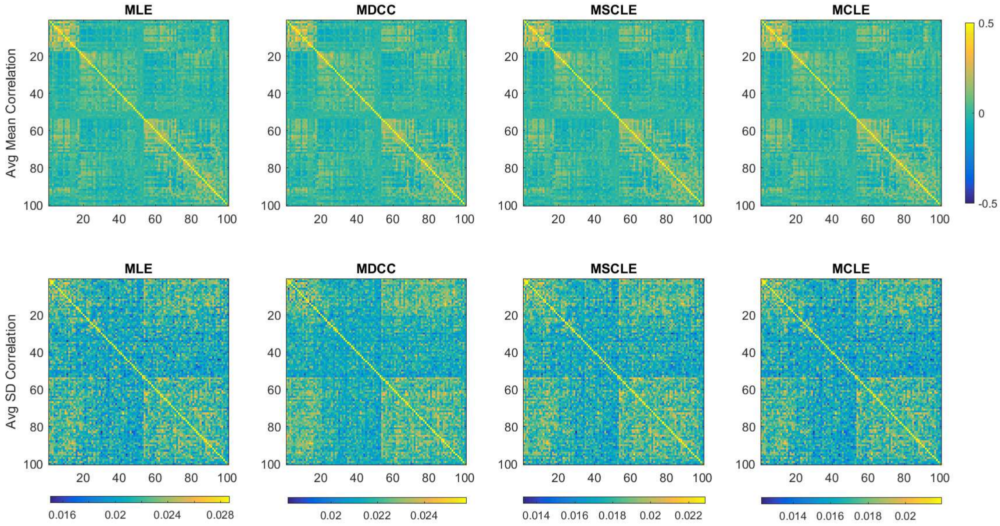

Figure 4.15: Application to HCP data. Estimates of the dynamic correlations for all edges computed simultaneously in a single DCC model using the four different methods (MLE, MDCC, MSCLE and MCLE).The average and the standard deviation of the estimated dynamic correlation of each edge are averaged across all subjects/scans, and displayed in matrix form.

Table 4.3: The average run time per subject/scan in the HCP data application

Run Time (second)	MLE	MDCC	MSCLE	MCLE
Single model for all pairs	213.0	23.3	24.2	1155.0
Separate bivariate models	1841.7	529.4	–	–

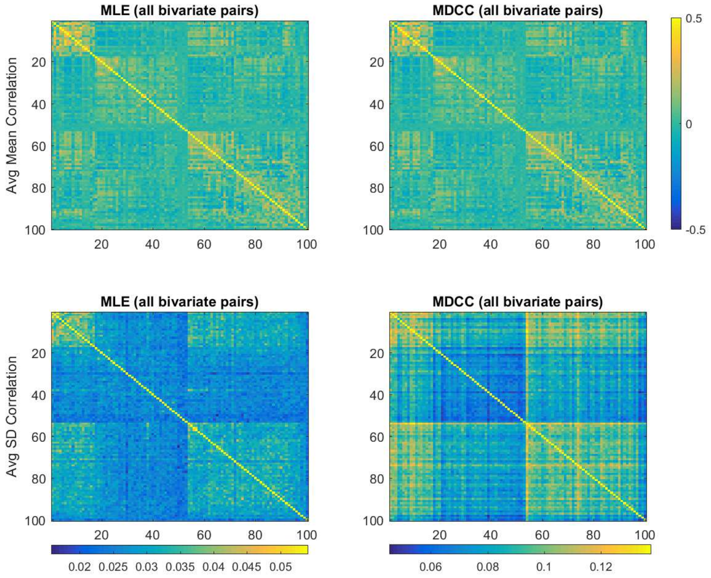

Figure 4.16: Application to HCP data. Estimates of the dynamic correlation for each edge computed separately with bivariate DCC models using MLE and MDCC. The average and the standard deviation of the estimated dynamic correlation of each edge are averaged across all subjects/scans, and displayed in matrix form.

4.6 Discussion

This paper introduced the MDCC method for estimating high-dimensional time-varying correlation/covariance matrices for multivariate time series, based upon using the method of moment equation and a fixed-point iteration-fitting algorithm. The performance of the proposed estimator is compared with alternative estimators in a series of simulations as well as empirical application to rs-fMRI data. The results demonstrated the effectiveness of the MDCC estimator, and also suggest the proper usage of various different alternative methods, including the standard quasi-likelihood approach (Engle, 2002a), as well as a recently developed composite likelihood approach (Pakel et al., 2014).

Our results show that in the low-dimensional setting, the MDCC procedure may lead to loss in efficiency compared to the original quasi-likelihood approach. However, it is easy to implement and scales well with high-dimensional data, even when the dimensions are larger than the length of time series. Thus, it provides an intriguing alternative to the composite likelihood based approach for dealing with high-dimensional time series. The relevant benefit of the MDCC compared to the composite likelihood methods ultimately depends on the situation. Here MDCC performs better when fluctuations differ between different pairs of time courses extracted from the data, while the composite likelihood methods perform better when the fluctuations are more uniform across pairs.

Future work may explore the robustness of these results with respect to the distribution of the residual errors, as well as to the presence of outliers or spikes, which commonly appear in fMRI data. Another possible direction for future investigation includes the development of a hybrid algorithm that combines the strength of both MDCC and MSCLE estimators, i.e. using MSCLE to focus on the dynamic correlation for a smaller subset of more important edges while using MDCC to take care of the remaining edges. Yet another possible extension involves the incorporation of an unsupervised clustering step within the DCC model, in order to separate ROIs with different time-varying patterns in FC, and thus allow for the use of different DCC parameters to model the dynamic FC for nodes within each cluster.

Chapter 5

Conclusion

In this thesis we have presented several new statistical methods for assessing dynamic functional connectivity in fMRI data. This is a problem that has attracted increased interest in the past couple of years as identification of dynamic FC estimation methods and summary metrics that maximize reliability has been shown to be critical for providing insight into brain function. Much research remains in the area, and we believe that in the coming years this will develop into one of the primary research areas related to fMRI data. Particular areas of research include the investigation of the role of autocorrelation in dynamic FC, the development of summary statistics for describing dynamic FC using a few summary metrics, and the development of inferential tests for assessing the presence of dynamic FC. It is our sincere hope that the work contained in this thesis provides fundamental building blocks required for moving this research forward.

A1 Appendix to Chapter 2

A1.1 Minimum partition length

We need to calculate a minimum partition length Δ to control the type II error based on a pre-specified bound β. Consider two time series, each of length Δ. Denote the test statistic $t_{stat} = \frac{\delta x}{\sqrt{\frac{2}{\Delta}}s}$, where s represents the pooled variance. Under the null hypothesis, t_{stat} follows a Student's t-distribution with $2\Delta - 2$ degrees of freedom, and we reject H_0 if $|t_{stat}| \geq t_{1-\alpha/2}(2\Delta - 2)$.

If the alternative hypothesis H_1 is true, and the actual difference in mean between the two groups is $\delta\mu$, then the statistic $t' = \frac{\delta x - \delta\mu}{\sqrt{\frac{2}{\Delta}}s}$ follows a Student's t-distribution with $2\Delta - 2$ degrees of freedom. Without loss of generality, assume that $\delta\mu > 0$. Then the type II error of this hypothesis test satisfies:

$$Pr(|\delta x|) \leq s \cdot t_{1-\alpha/2}(2\Delta - 2) \approx Pr(t' \leq t_{1-\alpha/2}(2\Delta - 2) - \frac{\delta\mu}{\sqrt{\frac{2}{\Delta}}s}) \leq \beta$$

In practice, we set the effect size as $\frac{\delta\mu}{s} = 1$ and since we are comparing time courses from J regions we use Bonferroni correction to set $\alpha \rightarrow \alpha/J$ and $\beta \rightarrow \beta/J$. Beginning at $\Delta = 10$, if $Pr(t' \leq t_{1-\alpha/2M}(2\Delta - 2) - \sqrt{\frac{\Delta}{2}})$ is larger than β, increase Δ by 1 until the above equation is satisfied.

A1.2 Jackknife resampling

The jackknife is a useful technique for variance estimation. It 'bootstraps' the estimator by systematically leaving out each observation and re-calculating the estimate. Suppose we have a sequence of data $\{X_t\}_{1\leq t\leq T}$, and we want to estimate the variance of an estimator:

$$\hat{\theta} = \frac{1}{T}\sum_t X_t$$

First we calculate the jackknife estimate of θ as

$$\theta_{Jack} = \frac{1}{T}\sum_t \tilde{\theta}_t$$

where $\tilde{\theta}_t$ is the estimator for a subsample omitting the t^{th} observation,

$$\tilde{\theta}_t = \frac{1}{T-1}\sum_{s\neq t} X_s$$

Hence,

$$\begin{aligned}\theta_{Jack} &= \frac{1}{T}\sum_t \frac{1}{T-1}(\sum_{s=1}^{T} X_s - X_t) \\ &= \frac{1}{T-1}\sum_t \big(\frac{1}{T}\sum_{s=1}^{T} X_s\big) - \frac{1}{T(T-1)}\sum_t X_t \\ &= \frac{T}{T-1}\hat{\theta} - \frac{1}{T-1}\hat{\theta} = \hat{\theta}\end{aligned} \tag{A.1.1}$$

Now calculate an estimate of the variance of $\hat{\theta}$ using the jackknife technique:

$$\begin{aligned} Var(\hat{\theta}) &= \frac{T-1}{T}\sum_t(\tilde{\theta}-\theta_{Jack})^2 \\ &= \frac{T-1}{T}\sum_t\big(\frac{1}{T-1}(\sum_{s=1}^{T}X_s - X_t)-\hat{\theta}\big)^2 \\ &= \frac{T-1}{T}\sum_t\big(\frac{T}{T-1}\hat{\theta}-\frac{1}{T-1}X_t-\hat{\theta}\big)^2 \qquad \text{(A.1.2)} \\ &= \frac{T-1}{T}\sum_t\frac{1}{(T-1)^2}(X_t-\hat{\theta})^2 \\ &= \frac{1}{(T-1)T}\sum_{t=1}^{T}(X_t-\hat{\theta})^2 \end{aligned}$$

Applying the result to Eq.2.1, we can estimate the variance of $\hat{\Sigma}_{ij}$ as

$$Var(\hat{\Sigma}_{ij}) = \frac{1}{(T-1)T}\sum_{1\le t\le T}(X_t^{ij}-\hat{\Sigma}_{ij})^2 \approx \frac{1}{T}\delta_{ij}^2$$

which is similar to that obtained using the central limit theorem.

A1.3 Simulation Setting

Below is a more detailed list of simulation studies, including the exact value of precision matrices used in simulation 2 to 6.

- **Simulation 1**

  Description: The data is white noise with no connectivity change points.

  Size: $N = 20$, $T = 1000$, $p = 20$

- **Simulation 2**

  Description: There are two change points at times 200 and 400. Spikes are imposed onto the time series, imitating a common artifact found in fMRI data. For each subject there are 5 randomly placed spikes, each with magnitude 15.

  Size: $N = 20$, $T = 1000$, $p = 20$

  Dependency Structure:

$$\begin{aligned} t \in [1, 200] : &\qquad (3, 14) = 0.3, (3, 9) = 0.6, (9.14) = 0.4 \\ t \in (200, 400] : &\qquad (1, 6) = 0.7, (6, 14) = 0.5, (1, 19) = 0.6 \\ t \in (400, 600] : &\qquad (3, 10) = 0.7, (3, 13) = 0.6, (3, 20) = 0.4, \\ &\qquad (10, 20) = 0.1, (13, 20) = 0.1 \end{aligned}$$

- **Simulation 3**

  Description: There are three change points at times 125, 500 and 750.

  Size: $N = 15$, $T = 1000$, $p = 20$

Dependency Structure:

$$\begin{aligned}
t \in [1, 125]: &\quad (2, 8) = 0.7, (8, 17) = 0.5, (2, 17) = 0.2 \\
t \in (125, 500]: &\quad (6, 14) = 0.1, (1, 6) = 0.3, (1, 18) = 0.2, \\
&\quad (1, 14) = 0.3, (6, 18) = 0.4 \\
t \in (500, 750]: &\quad (3, 8) = 0.5, (8, 13) = 0.5, (13, 19) = 0.4, (3, 19) = 0.4, \\
&\quad (3, 13) = 0.1, (8, 19) = 0.2 \\
t \in (750, 1000]: &\quad (5, 11) = 0.8
\end{aligned}$$

- **Simulation 4**

  Description: There is a single change point at time 100.

  Size: $N = 25$, $T = 200$, $p = 5$

  Dependency Structure:

$$\begin{aligned}
t \in [1, 100]: &\quad (1, 3) = 0.7, (3, 5) = 0.6, (1, 5) = 0.3, \\
&\quad (3, 4) = 0.2, (4, 5) = 0.2, (1, 4) = 0.1 \\
t \in (100, 200]: &\quad (1, 2) = -0.1, (1, 5) = -0.2, (2, 5) = 0.4
\end{aligned}$$

- **Simulation 5**

  Description: There are five change points at times 200, 300, 500, 600, and 800.

  Size: $N = 20$, $T = 1000$, $p = 20$

Dependency Structure:

$$
\begin{aligned}
t \in [1, 200]: &\quad (2, 14) = 0.8 \\
t \in (200, 300]: &\quad (2, 14) = 0.4, (3, 9) = 0.3, (9, 18) = 0.4, (3, 18) = 0.3 \\
t \in (300, 500]: &\quad (3, 9) = 0.7, (3, 18) = 0.5, (9, 18) = 0.3 \\
t \in (500, 600]: &\quad (2, 19) = 0.4, (3, 18) = 0.3, (2, 13) = 0.5, \\
&\quad (6, 13) = 0.2, (9, 18) = 0.3 \\
t \in (600, 800]: &\quad (2, 6) = 0.6, (6, 19) = 0.5, (2, 19) = 0.3, (2, 13) = 0.5 \\
t \in (800, 1000]: &\quad (1, 11) = 0.9
\end{aligned}
$$

- **Simulation 6**

  Description: There are four change points at times 200, 400, 600, and 800.

  Size: $N = 20$, $T = 1000$, $p = 20$

  Dependency Structure:

$$
\begin{aligned}
t \in [1, 200]: &\quad (1, 5) = 0.8, (5, 10) = 0.3, (10, 15) = 0.5 \\
t \in (200, 400]: &\quad (2, 9) = 0.6, (9, 18) = 0.3 \\
t \in (400, 600]: &\quad (3, 6) = 0.4, (6, 13) = 0.3, (13, 19) = 0.2 \\
t \in (600, 800]: &\quad (4, 8) = 0.7, (8, 15) = 0.3, (15, 20) = 0.6 \\
t \in (800, 1000]: &\quad (2, 14) = 0.5
\end{aligned}
$$

A1.4 DCD algorithm pseudocode

The procedure for performing the DCD algorithm is listed below:

1. Take the input parameters α, β, η, and calculate the minimum partition length Δ as described in Appendix A1.1.
2. Consider the full multivariate time series with length T, calculate the sparsity structure of its multivariate normal distribution parameters as described in Section
3. For each value of t ranging from Δ to $T - \Delta$, partition the time series into two subsequences $\{1 : t\}$ and $\{t + 1 : T\}$, calculate the sparsity structure of
4. Find the time point which produces the largest increase in combined likelihood function, perform the hypothesis test described in Eq. 2.4 to determine
5. Apply Steps (2) - (4) recursively to each partition until no further change points are found.

A2 Appendix to Chapter 3

Let $\gamma = (\omega_1, \alpha_1, \beta_1, \omega_2, \alpha_2, \beta_2)$ represent the parameters of the two GARCH models and $\phi = (\theta_1, \theta_2)$ the parameters of the correlation process. It can be shown Engle and Sheppard (2001); Engle (2002a) that the log-likelihood function for the DCC model can be expressed as the sum of a volatility term (corresponding to the individual

GARCH models for each of the two time series) and a correlation term. The log-likelihood can be written:

$$\mathcal{L}(\gamma, \phi) = \mathcal{L}_V(\gamma) + \mathcal{L}_C(\gamma, \phi) \tag{A.2.3}$$

where

$$\mathcal{L}_V(\gamma) = -\frac{1}{2}\sum_{t=1}^{T}(2log(2\pi) + \log|D_t|^2 + r_t^T D_t^{-2} r_t) \tag{A.2.4}$$

$$\mathcal{L}_C(\gamma, \phi) = -\frac{1}{2}\sum_{t=1}^{T}\log|R_t| + \epsilon_t^T R_t^{-1}\epsilon_t - \epsilon_t^T\epsilon_t \tag{A.2.5}$$

The parameters (γ, ϕ) can be estimated using a two-stage approach. First, the parameters of the volatility model are estimated by computing:

$$\hat{\gamma} = \underset{\gamma}{\operatorname{argmax}}\, \mathcal{L}_V(\gamma). \tag{A.2.6}$$

Thereafter, the parameters of the correlation model can be estimated by plugging $\hat{\gamma}$ into $\mathcal{L}_C(\gamma, \phi)$ and computing:

$$\hat{\phi} = \underset{\gamma}{\operatorname{argmax}}\, \mathcal{L}_C(\hat{\gamma}, \phi). \tag{A.2.7}$$

Both solutions can be obtained using standard maximization techniques.

Under certain very general conditions Engle and Sheppard (2001), the estimates obtained using this two-stage approach are consistent and asymptotically normal with variance $A^{-1}BA^{-1}$ where

$$A = \begin{pmatrix} \nabla_{\gamma\gamma} \log f_V(\gamma_0) & 0 \\ \nabla_{\gamma\phi} \log f_C(\gamma_0, \phi_0) & \nabla_{\phi\phi} \log f_C(\gamma_0, \phi_0) \end{pmatrix} \tag{A.2.8}$$

and

$$B = var[\sum_{t=1}^{T} \{T^{-1/2} \nabla_{\gamma}^{T} \log f_V(r_t, \gamma_0), T^{-1/2} \nabla_{\phi}^{T} \log f_C(r_t, \gamma_0, \phi_0)]. \tag{A.2.9}$$

Here ∇_x represents a partial derivative with respect to a parameter x, and f_V and f_C are the likelihood functions for the volatility and correlation portion of the likelihood function, respectively.

These variances can be approximated using finite difference methods and used to approximate a distribution for the model parameters. Using Monte Carlo sampling, we can randomly generate a number of draws for these distributions, calculate the dynamic correlations for each set of parameters, and use this to construct confidence intervals.

A3 Appendix to Chapter 4

1) Simplified formulation of the design matrix in Eq. 4.12.

After we obtain ϵ_t^* and thus Y_t, we can calculate the empirical covariance matrix $S = \frac{1}{T}\sum_{t=1}^{T}\epsilon_t^*\epsilon_t^{*'}$, and subtract $vec(S)$ (treated as a fixed constant vector) from both sides of Eq. 4.12. This gives us:

$$E[Y_t - vec(S)|\mathcal{F}_{t-1}] = \mu_t - vec(S) = \phi_1 \cdot \big(Y_{t-1} - vec(S)\big) + \phi_2 \cdot \big(\mu_{t-1} - vec(S)\big) \quad \text{(A.3.10)}$$

Now let

$$\tilde{Y}_t = Y_t - vec(S), \qquad \tilde{\mu}_t = \mu_t - vec(S), \qquad \tilde{X}_t = \big[\tilde{Y}_{t-1}, \tilde{\mu}_{t-1}\big]$$

then we can write

$$\tilde{\mu}_t = E[\tilde{Y}_t|\mathcal{F}_{t-1}] = \tilde{X}_t \cdot \phi, \qquad \phi = [\phi_1, \phi_2]^T$$

2) Solution to the first optimization problem in Eq. 4.13:

We can express the objective function as follows:

$$
\begin{aligned}
F(\phi;\{\epsilon_t\}_{1\le t\le T}) &= \underset{\phi\in\Phi}{argmin}\sum_{t=1}^{T}\left(Y_t-\mu_t\right)^2 = \underset{\phi\in\Phi}{argmin}\sum_{t=1}^{T}\left(\tilde{Y}_t-\tilde{\mu}_t\right)^2 \\
&= \underset{\phi\in\Phi}{argmin}\sum_{t=1}^{T}\left(\tilde{Y}_t-\tilde{X}_t\cdot\phi\right)^2 \\
&= \underset{\phi\in\Phi}{argmin}\left\{\phi^T(\sum_{t=1}^{T}\tilde{X}_t^T\tilde{X}_t)\phi-2(\sum_{t=1}^{T}\tilde{Y}_t^T\tilde{X}_t)\phi\right\} \\
&= \underset{\phi\in\Phi}{argmin}\left\{\phi^T A\phi-2b^T\phi\right\}
\end{aligned}
\tag{A.3.11}
$$

where

$$
A=\sum_{t=1}^{T}\tilde{X}_t^T\tilde{X}_t, \qquad b=\sum_{t=1}^{T}\tilde{X}_t^T\tilde{Y}_t
$$

This is a standard convex optimization problem. The solution of $F(\phi;\{\epsilon_t\}_{1\le t\le T})$ is either $A^{-1}b$ (if $\phi\in\Phi$), or in one of the boundaries of the 2-dimensional triangle Φ. With a 2-dimensional matrix A and vector b, the solution can be computed in $O(1)$ time.

3) Solution to the second optimization problem in Eq. 4.14.

We are interested in finding the root of:

$$
G(\phi) := F(\phi;\{\epsilon_t\}_{1\le t\le T})-\phi \tag{A.3.12}
$$

There are a number of root-finding methods that can be applied to this problem. We start from a random initial point $\phi_0\in\Phi$, then perform several rounds of fixed point iteration as specified below:

$$\phi_{n+1} = F(\phi_n; \{\epsilon_t\}_{1 \le t \le T})$$

Empirical evidence tells us that regardless of the initial point, this procedure will quickly move ϕ_n towards the truth, but may become stuck in a limiting cycle when it's sufficiently close to the root. For this reason we use the procedure described in the Methods section, which uses a random linear combination of ϕ_n and $F(\phi_n; \{\epsilon_t\}_{1 \le t \le T})$ to update ϕ_{n+1}. In our empirical studies, this algorithm achieves fast convergence when the true parameter is not too close to the boundary. In practice this procedure outperforms all algorithms implemented in Matlab's "fmincon" function in terms of both speed and accuracy.

Bibliography

Aielli, G. P. (2013). Dynamic conditional correlation: on properties and estimation. *Journal of Business & Economic Statistics*, 31(3):282–299.

Allen, E. A., Damaraju, E., Plis, S. M., Erhardt, E. B., Eichele, T., and Calhoun, V. D. (2012). Tracking whole-brain connectivity dynamics in the resting state. *Cerebral Cortex*.

Andersen, T. G., Bollerslev, T., Diebold, F. X., and Labys, P. (2003). Modeling and forecasting realized volatility. *Econometrica*, 71(2):579–625.

Bauwens, L., Laurent, S., and Rombouts, J. V. (2006). Multivariate garch models: a survey. *Journal of applied econometrics*, 21(1):79–109.

Beckmann, C. F. and Smith, S. M. (2004). Probabilistic independent component analysis for functional magnetic resonance imaging. *Medical Imaging, IEEE Transactions on*, 23(2):137–152.

Behzadi, Y., Restom, K., Liau, J., and Liu, T. T. (2007). A component based noise correction method (compcor) for bold and perfusion based fmri. *Neuroimage*, 37(1):90–101.

Biswal, B., Zerrin Yetkin, F., Haughton, V. M., and Hyde, J. S. (1995). Functional connectivity in the motor cortex of resting human brain using echo-planar mri. *Magnetic resonance in medicine*, 34(4):537–541.

Biswal, B. B., Kylen, J. V., and Hyde, J. S. (1997). Simultaneous assessment of flow and bold signals in resting-state functional connectivity maps. *NMR in Biomedicine*, 10(45):165–170.

Blondel, V. D., Guillaume, J.-L., Lambiotte, R., and Lefebvre, E. (2008). Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008.

Bollerslev, T. (1986). Generalized autoregressive conditional heteroskedasticity. *Journal of econometrics*, 31(3):307–327.

Cai, T. and Liu, W. (2011). Adaptive thresholding for sparse covariance matrix estimation. *Journal of the American Statistical Association*, 106(494):672–684.

Calhoun, V. D., Miller, R., Pearlson, G., and Adal, T. (2014). The chronnectome: time-varying connectivity networks as the next frontier in fMRI data discovery. *Neuron*, 84(2):262–274.

Caporin, M. and McAleer, M. (2012). Do we really need both bekk and dcc? a tale of two multivariate garch models. *Journal of Economic Surveys*, 26(4):736–751.

Chang, C. and Glover, G. H. (2010). Time–frequency dynamics of resting-state brain connectivity measured with fmri. *Neuroimage*, 50(1):81–98.

Chang, C., Liu, Z., Chen, M. C., Liu, X., and Duyn, J. H. (2013). Eeg correlates of time-varying bold functional connectivity. *Neuroimage*.

Cribben, I., Haraldsdottir, R., Atlas, L. Y., Wager, T. D., and Lindquist, M. A. (2012). Dynamic connectivity regression: Determining state-related changes in brain connectivity. *NeuroImage*, 61(4):907–920.

Cribben, I., Wager, T. D., and Lindquist, M. A. (2013). Detecting functional connectivity change points for single-subject fmri data. *Frontiers in computational neuroscience*, 7.

Damaraju, E., Allen, E., Belger, A., Ford, J., McEwen, S., Mathalon, D., Mueller, B., Pearlson, G., Potkin, S., Preda, A., et al. (2014). Dynamic functional connectivity analysis reveals transient states of dysconnectivity in schizophrenia. *NeuroImage: Clinical*, 5:298–308.

Engle, R. (2002a). Dynamic conditional correlation: A simple class of multivariate generalized autoregressive conditional heteroskedasticity models. *Journal of Business & Economic Statistics*, 20(3):339–350.

Engle, R. (2002b). New frontiers for arch models. *Journal of Applied Econometrics*, 17(5):425–446.

Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of united kingdom inflation. *Econometrica: Journal of the Econometric Society*, pages 987–1007.

Engle, R. F., Ledoit, O., and Wolf, M. (2016). Large dynamic covariance matrices. *working paper*.

Engle, R. F., Shephard, N., and Sheppard, K. (2008). Fitting vast dimensional time-varying covariance models. *Working paper*.

Engle, R. F. and Sheppard, K. (2001). Theoretical and empirical properties of dynamic conditional correlation multivariate garch. Technical report, National Bureau of Economic Research.

Filippini, N., MacIntosh, B. J., Hough, M. G., Goodwin, G. M., Frisoni, G. B., Smith, S. M., Matthews, P. M., Beckmann, C. F., and Mackay, C. E. (2009). Distinct patterns of brain activity in young carriers of the apoe-ε4 allele. *Proceedings of the National Academy of Sciences*, 106(17):7209–7214.

Fox, M. D. and Raichle, M. E. (2007). Spontaneous fluctuations in brain activity observed with functional magnetic resonance imaging. *Nature Reviews Neuroscience*, 8(9):700–711.

Friedman, J., Hastie, T., and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441.

Friston, K., Buechel, C., Fink, G., Morris, J., Rolls, E., and Dolan, R. (1997). Psychophysiological and modulatory interactions in neuroimaging. *Neuroimage*, 6(3):218–229.

Friston, K. J. (1994). Functional and effective connectivity in neuroimaging: a synthesis. *Human brain mapping*, 2(1-2):56–78.

Friston, K. J. (2011). Functional and effective connectivity: a review. *Brain connectivity*, 1(1):13–36.

Ginestet, C. E. and Simmons, A. (2011). Statistical parametric network analysis of functional connectivity dynamics during a working memory task. *Neuroimage*, 55(2):688–704.

Glasser, M. F., Sotiropoulos, S. N., Wilson, J. A., Coalson, T. S., Fischl, B., Andersson, J. L., Xu, J., Jbabdi, S., Webster, M., Polimeni, J. R., et al. (2013). The minimal preprocessing pipelines for the human connectome project. *Neuroimage*, 80:105–124.

Griffanti, L., Salimi-Khorshidi, G., Beckmann, C. F., Auerbach, E. J., Douaud, G., Sexton, C. E., Zsoldos, E., Ebmeier, K. P., Filippini, N., Mackay, C. E., et al.

(2014). Ica-based artefact removal and accelerated fmri acquisition for improved resting state network imaging. *NeuroImage*, 95:232–247.

Hafner, C. M. and Reznikova, O. (2012). On the estimation of dynamic conditional correlation models. *Computational Statistics & Data Analysis*, 56(11):3533–3545.

Handwerker, D. A., Roopchansingh, V., Gonzalez-Castillo, J., and Bandettini, P. A. (2012). Periodic changes in fmri connectivity. *Neuroimage*.

Hansen, P. R. and Lunde, A. (2005). A forecast comparison of volatility models: does anything beat a garch (1, 1)? *Journal of applied econometrics*, 20(7):873–889.

Hutchison, R. M. and Morton, J. B. (2015). Tracking the brain's functional coupling dynamics over development. *The Journal of Neuroscience*, 35(17):6849–6859.

Hutchison, R. M., Womelsdorf, T., Allen, E. A., Bandettini, P. A., Calhoun, V. D., Corbetta, M., Penna, S. D., Duyn, J., Glover, G., Gonzalez-Castillo, J., et al. (2013). Dynamic functional connectivity: Promises, issues, and interpretations. *NeuroImage*.

Jeong, J., Gore, J. C., and Peterson, B. S. (2001). Mutual information analysis of the eeg in patients with alzheimer's disease. *Clinical Neurophysiology*, 112(5):827–835.

Jones, D. T., Vemuri, P., Murphy, M. C., Gunter, J. L., Senjem, M. L., Machulda, M. M., Przybelski, S. A., Gregg, B. E., Kantarci, K., Knopman, D. S., et al.

(2012). Non-stationarity in the “resting brain’s” modular architecture. *PloS one*, 7(6):e39731.

Kiviniemi, V., Vire, T., Remes, J., Elseoud, A. A., Starck, T., Tervonen, O., and Nikkinen, J. (2011). A sliding time-window ica reveals spatial variability of the default mode network in time. *Brain connectivity*, 1(4):339–347.

Kwong, K. K., Belliveau, J. W., Chesler, D. A., Goldberg, I. E., Weisskoff, R. M., Poncelet, B. P., Kennedy, D. N., Hoppel, B. E., Cohen, M. S., and Turner, R. (1992). Dynamic magnetic resonance imaging of human brain activity during primary sensory stimulation. *Proceedings of the National Academy of Sciences*, 89(12):5675–5679.

Landman, B. A., Huang, A. J., Gifford, A., Vikram, D. S., Lim, I. A. L., Farrell, J. A., Bogovic, J. A., Hua, J., Chen, M., Jarso, S., et al. (2011). Multi-parametric neuroimaging reproducibility: a 3-t resource study. *Neuroimage*, 54(4):2854–2866.

Lebo, M. J. and Box-Steffensmeier, J. M. (2008). Dynamic conditional correlations in political science. *American Journal of Political Science*, 52(3):688–704.

Leonardi, N. and Van De Ville, D. (2015). On spurious and real fluctuations of dynamic functional connectivity during rest. *Neuroimage*, 104:430–436.

Lepskii, O. (1990). On a problem of adaptive estimation in gaussian white noise. *Theory of Probability & Its Applications*, 35(3):454–466.

Lindquist, M. and Wager, T. D. (2005). Application of change-point theory to modeling state-related activity in fmri. *Applied Data Analytic Techniques for "Turning Points Research*.

Lindquist, M. A. (2008). The statistical analysis of fmri data. *Statistical Science*, 23(4):439–464.

Lindquist, M. A. and Mejia, A. (2015). Zen and the art of multiple comparisons. *Psychosomatic medicine*, 77(2):114–125.

Lindquist, M. A., Waugh, C., and Wager, T. D. (2007). Modeling state-related fmri activity using change-point theory. *Neuroimage*, 35(3):1125–1141.

Lindquist, M. A., Xu, Y., Nebel, M. B., and Caffo, B. S. (2014). Evaluating dynamic bivariate correlations in resting-state fmri: A comparison study and a new approach. *NeuroImage*, 101:531–546.

Ogawa, S., Tank, D. W., Menon, R., Ellermann, J. M., Kim, S. G., Merkle, H., and Ugurbil, K. (1992). Intrinsic signal changes accompanying sensory stimulation: functional brain mapping with magnetic resonance imaging. *Proceedings of the National Academy of Sciences*, 89(13):5951–5955.

Ombao, H. and Van Bellegem, S. (2008). Evolutionary coherence of nonstationary signals. *Signal Processing, IEEE Transactions on*, 56(6):2259–2266.

Pakel, C., Shephard, N., Sheppard, K., and Engle, R. F. (2014). Fitting vast dimensional time-varying covariance models. Technical report, Working paper.

Pang, C.-m. (1997). Generalized method of moments estimation of generalized autoregressive conditional heteroskedastic models. *HKUST Electronic Theses*.

Purdon, P. L., Solo, V., Weisskoff, R. M., and Brown, E. N. (2001). Locally regularized spatiotemporal modeling and model comparison for functional mri. *NeuroImage*, 14(4):912–923.

Rashid, B., Damaraju, E., Pearlson, G. D., and Calhoun, V. D. (2014). Dynamic connectivity states estimated from resting fmri identify differences among schizophrenia, bipolar disorder, and healthy control subjects. *Frontiers in human neuroscience*, 8:897.

Robinson, L. F., Wager, T. D., and Lindquist, M. A. (2010). Change point estimation in multi-subject fmri studies. *Neuroimage*, 49(2):1581–1592.

Salimi-Khorshidi, G., Douaud, G., Beckmann, C. F., Glasser, M. F., Griffanti, L., and Smith, S. M. (2014). Automatic denoising of functional mri data: combining independent component analysis and hierarchical fusion of classifiers. *Neuroimage*, 90:449–468.

Sheppard, K. (2012). Forecasting high dimensional covariance matrices. *Handbook of Volatility Models and Their Applications*, pages 103–125.

Smith, S. M., Beckmann, C. F., Andersson, J., Auerbach, E. J., Bijsterbosch, J., Douaud, G., Duff, E., Feinberg, D. A., Griffanti, L., Harms, M. P., et al. (2013). Resting-state fmri in the human connectome project. *NeuroImage*, 80:144–168.

Sporns, O., Tononi, G., and Kötter, R. (2005). The human connectome: a structural description of the human brain. *PLoS computational biology*, 1(4):e42.

Sun, F. T., Miller, L. M., and D'Esposito, M. (2004). Measuring interregional functional connectivity using coherence and partial coherence analyses of fmri data. *Neuroimage*, 21(2):647–658.

Tagliazucchi, E., Von Wegner, F., Morzelewski, A., Brodbeck, V., and Laufs, H. (2012). Dynamic bold functional connectivity in humans and its electrophysiological correlates. *Frontiers in human neuroscience*, 6.

Thompson, G. J., Magnuson, M. E., Merritt, M. D., Schwarb, H., Pan, W.-J., McKinley, A., Tripp, L. D., Schumacher, E. H., and Keilholz, S. D. (2012). Short-time windows of correlation between large-scale functional brain networks predict vigilance intraindividually and interindividually. *Human brain mapping*.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288.

Tsay, R. S. (2006). Multivariate volatility models. *Lecture Notes-Monograph Series*, pages 210–222.

Van Essen, D. C., Smith, S. M., Barch, D. M., Behrens, T. E., Yacoub, E., Ugurbil, K., Consortium, W.-M. H., et al. (2013). The wu-minn human connectome project: an overview. *Neuroimage*, 80:62–79.

Varoquaux, G., Gramfort, A., Poline, J.-B., and Thirion, B. (2010). Brain covariance selection: better individual functional connectivity models using population prior. In *Advances in Neural Information Processing Systems*, pages 2334–2342.

Wager, T. D., Waugh, C. E., Lindquist, M., Noll, D. C., Fredrickson, B. L., and Taylor, S. F. (2009). Brain mediators of cardiovascular responses to social threat: part i: Reciprocal dorsal and ventral sub-regions of the medial prefrontal cortex and heart-rate reactivity. *Neuroimage*, 47(3):821–835.

CURRICULUM VITAE

Yuting Xu

xuyuting@jhu.edu

615 N. Wolfe St. E3040

Baltimore, MD 21205

http://www.biostat.jhsph.edu/∼yuxu

Education

2012 - 2017 **Johns Hopkins Bloomberg School of Public Health**, Baltimore, Maryland, USA

Ph.D. in Biostatistics

Thesis title: *Dynamic Functional Connectivity in Functional Magnetic Resonance Imaging Data*

Advisor: Dr. Martin A. Lindquist

2015 - 2016 **Johns Hopkins University**, Baltimore, Maryland, USA

M.S.E. in Computer Science

2008 - 2012 **Tsinghua University**, Beijing, China

B.S. in Mathematics and Physics

CURRICULUM VITAE

Professional Experience

06/2016 - 08/2016	**Biostatistics Intern**
	Biometrics Research, Merck & Co., Inc., Rahway, NJ, U.S.A.
09/2013 - 01/2017	**Research Assistant**
	Department of Biostatistics, Johns Hopkins University.
	Advisor: Martin A. Lindquist

Publications

Xu, Y., & Lindquist, M. A. (2015). Dynamic connectivity detection: an algorithm for determining functional connectivity change points in fMRI data. Frontiers in neuroscience, 9.

Lindquist, M. A., **Xu, Y.**, Nebel, M. B., & Caffo, B. S. (2014). Evaluating dynamic bivariate correlations in resting-state fMRI: A comparison study and a new approach. Neuroimage, 101, 531-546.

Xu, Y., & Lindquist, M. A. (2016). Moment Estimator for GARCH-DCC Model: an algorithm for dynamic functional connectivity in high dimensional fMRI data. *In preparation.*

Xu, Y., Ma, J., Liaw A., Sheridan, R., Svetnik, V. (2016). Demystifying Muti-task Deep Neural Networks (DNN) for Quantitative Structure-Activity Relationships

(QSAR). *In preparation.*

Xu, Y., & Lan, Y. (2012). The N-leap method for stochastic simulation of coupled chemical reactions. The Journal of chemical physics, 137(20), 204103.

Li, Q., Wennborg, A., Aurell, E., Dekel, E., Zou, J. Z., **Xu, Y.**, Huang, S., Ernberg, I. (2016). Dynamics inside the cancer cell attractor reveal cell heterogeneity, limits of stability, and escape. Proceedings of the National Academy of Sciences, 113(10), 2672-2677.

Chen, S., Liu, K., Yang, Y., **Xu, Y.**, Lee, S., Lindquist, M., ... & Vogelstein, J. T. (2016). An M-Estimator for Reduced-Rank High-Dimensional Linear Dynamical System Identification. Pattern Recognition Letters. (Accepted)

Yue, C., **Xu, Y.**, Chen, S., Goldsmith, J., Caffo, BS., Zipunnikov, V. (2016) Multilevel Binary Principal Component Analysis. *In preparation.*

Y. Webb-Vargas, S. Chen, A. Fisher, A. Mejia, **Y. Xu.**, C. Crainiceanu, B. Caffo, M. A. Lindquist. (2016). Big Data and Neuroimaging. Statistics in Biosciences (Accepted).

Professional Activities

PRESENTATIONS

2015.03	Penalized Clustering Using a Hidden Markov Random Field Model: Detecting State-Related Changes in Brain Connectivity. *ENAR Spring Meeting, Miami, FL*

POSTERS

2016.03	Multilevel Binary Principal Component Analysis. *ENAR Spring Meeting, Austin, TX*

Teaching Assistant

2016	Statistical Reasoning *I-II*, 140.611-612.
2013-2016	Statistical Methods in Public Health *I-II*, 140.621-622.
2013-2016	Statistical Methods in Public Health *III-IV*, 140.623-624.
2013-2014	Principles and Methods of Functional Neuroimaging *I-II*, 140.682-683.